

BAB IV

IMPLEMENTASI DAN EVALUASI

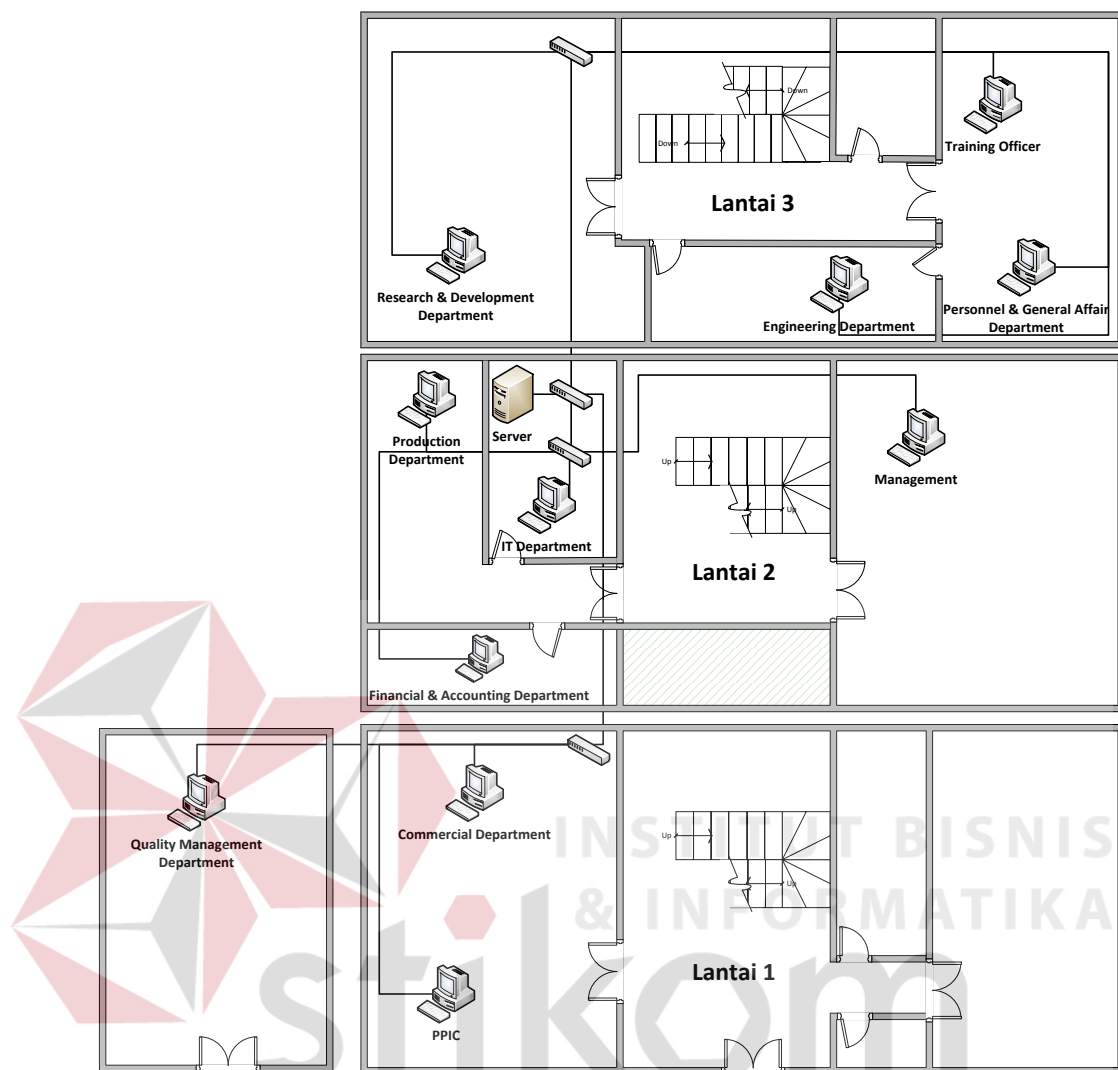
4.1 Implementasi

Implementasi program merupakan implementasi dari hasil analisis dan desain sistem yang telah dibuat sebelumnya. Dengan adanya implementasi ini, diharapkan dapat membantu perusahaan dalam melakukan analisis kebutuhan pelatihan yang lebih optimal sehingga pelaksanaan pelatihan terlaksana dengan efektif dan efisien. Sebelum melakukan implementasi, pengguna harus mempersiapkan kebutuhan-kebutuhan dari program yang akan diimplementasikan, yaitu berupa perangkat lunak dan kebutuhan perangkat keras seperti yang telah dijelaskan pada bab sebelumnya.

4.2 Konfigurasi Perangkat Keras dan Perangkat Lunak

Setelah aplikasi analisis kebutuhan pelatihan dirancang dan dibuat kemudian diimplementasikan dengan konfigurasi perangkat keras sesuai spesifikasi minimum yang telah ditentukan pada bab sebelumnya. Aplikasi tersebut dipasang ke setiap unit komputer dengan jumlah 12 unit yang terbagi dalam 11 komputer *client* dan 1 komputer *server*. Untuk konfigurasi jaringan akan menggunakan model *client-server* yang telah dijelaskan pada bab sebelumnya. Lebih jelasnya tentang konfigurasi jaringan akan digambarkan pada Gambar 4.1.

Dalam menjalankan Aplikasi Analisis Kebutuhan Pelatihan, dibutuhkan sebuah sistem operasi yang digunakan pada komputer *client* dan *server*. Komputer *client* akan menggunakan sistem operasi Windows 7 Professional. Sistem operasi pada komputer *server* adalah Windows Server 2000.



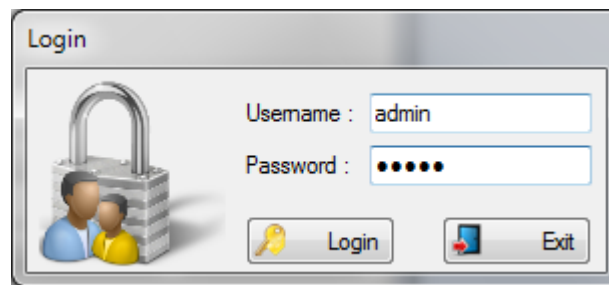
Gambar 4.1 Konfigurasi Jaringan *Client-Server*

4.3 Penjelasan Penggunaan Aplikasi

Aplikasi Analisis Kebutuhan Pelatihan Berbasis Kompetensi akan dijalankan sesuai dengan pembagian hak akses untuk setiap pengguna yang didaftarkan. Penjelasan berikut ini difokuskan pada fungsi-fungsi utama aplikasi sesuai dengan yang telah dispesifikasikan. Namun sebelumnya akan dijelaskan *form* yang digunakan oleh setiap pengguna dalam melakukan proses analisis kebutuhan pelatihan. Adapun *form* tersebut adalah sebagai berikut:

1. Form Login

Form login merupakan halaman yang ditampilkan pada saat pertama kali aplikasi dijalankan. *Form* ini berlaku bagi semua pengguna. Lebih jelasnya dapat dilihat pada Gambar 4.2.



Gambar 4.2 *Form Login*

Adapun penjelasan lebih detail mengenai *form login* dapat dilihat pada Tabel 4.1.

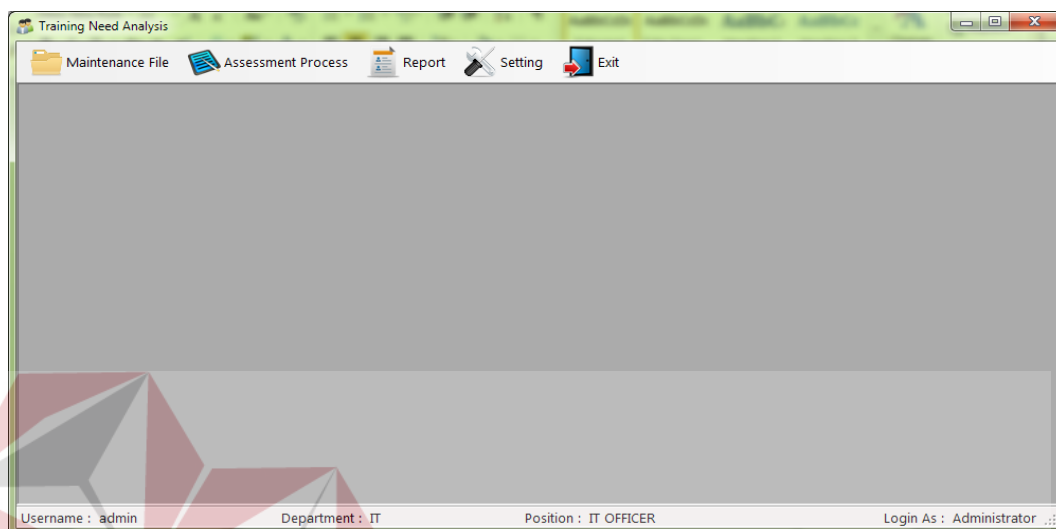
Tabel 4.1 Penjelasan *Form Login*

Fungsional	-	
Deskripsi	Sebelum masuk ke dalam aplikasi, pengguna harus memasukkan <i>username</i> dan <i>password</i> ke dalam sistem. Untuk mendapatkan hak akses pengguna, terlebih dahulu mendaftarkan diri ke bagian administrasi yang bertugas sebagai pemegang hak akses penuh terhadap aplikasi ini.	
Fungsi Tombol	<i>Login</i>	Digunakan setelah pengguna mengisi field " <i>Username</i> " dan " <i>Password</i> ". Setelah itu pengguna menekan tombol ini untuk masuk ke dalam aplikasi.
	<i>Exit</i>	Berfungsi untuk keluar dari <i>form login</i> , sekaligus membatalkan proses <i>login</i> ke dalam aplikasi.

2. Form Utama

Setelah berhasil melakukan *login*, maka *form* utama akan muncul dan akan dapat digunakan untuk melakukan proses yang diinginkan. Pada kasus ini akan

dibahas proses yang sesuai dengan spesifikasi untuk melakukan analisis kebutuhan pelatihan. Lebih jelas mengenai *form* utama dapat dilihat pada Gambar 4.3.



Gambar 4.3 *Form* Utama

Adapun penjelasan lebih detail mengenai *form* utama dapat dilihat pada Tabel 4.2.

Tabel 4.2 Penjelasan *Form* Utama

Fungsional	-	
Deskripsi	Pada <i>form</i> utama terdapat beberapa menu yang digunakan untuk masuk ke dalam form lainnya. Pada <i>form</i> ini juga terlihat ada profil pengguna sistem yang sedang menjalankan aplikasi.	
Fungsi Tombol	<i>Maintenance File</i>	Menu ini berfungsi untuk mengelola data induk yang akan digunakan dalam proses analisis kebutuhan pelatihan. Menu ini terdapat sub menu <i>Department</i> , <i>Employees</i> , <i>Competency</i> , <i>Standard Competency</i> .
	<i>Assessment Process</i>	Menu ini berfungsi untuk melakukan penilaian indikator perilaku pada setiap tingkat kompetensi. Penilaian tersebut digunakan untuk menentukan tingkat kompetensi yang dimiliki oleh karyawan. Menu ini terdapat sub menu <i>Employee Assessment</i> dan <i>Assessment Report</i> .

	<i>Report</i>	Menu ini digunakan untuk melihat dan mencetak laporan kebutuhan pelatihan pada karyawan. Menu ini terdapat sub menu <i>Department Analysis, Individual Analysis, Training Participants</i> dan <i>Training Need Priority</i> .
	<i>Setting</i>	Menu ini berfungsi untuk melakukan proses konfigurasi pada aplikasi. Konfigurasi yang dilakukan adalah konfigurasi terhadap nama <i>server</i> yang digunakan dalam mengakses tempat penyimpanan data (<i>database</i>) dan konfigurasi pada pengguna (<i>user</i>) dalam menjalankan aplikasi.
	<i>Exit</i>	Berfungsi untuk keluar dari aplikasi dan menghentikan proses yang akan dilakukan dalam proses melakukan analisis kebutuhan pelatihan.

Setelah dijelaskan mengenai beberapa *form* yang digunakan oleh semua pengguna, maka berikut ini akan dijelaskan mengenai *form* yang digunakan oleh tiap pengguna yang disesuaikan dengan fungsionalnya. Adapun penjelasan penggunaan aplikasi berdasarkan otorisasi pengguna adalah sebagai berikut:

4.3.1 Pengguna Sebagai *Department Head*

Department head hanya melakukan fungsional merencanakan pengembangan Karyawan. Fungsional ini terdapat 2 *form* yang digunakan untuk menentukan pelatihan yang dibutuhkan oleh karyawan, yaitu *form employee assessment* dan *form behavior indicator assessment*. Sebelum ditentukan pelatihan yang dibutuhkan, kompetensi karyawan harus dilakukan penilaian oleh *department head*. Penilaian tersebut adalah penilaian indikator perilaku pada setiap tingkat kompetensi. Setelah itu, tingkat kompetensi karyawan akan diketahui dan sistem akan melakukan penentuan kebutuhan pelatihan karyawan. Sesuai dengan aturan perusahaan mengenai penilaian kompetensi karyawan, *department head* melakukan penilaian pada setiap setahun sekali untuk digunakan

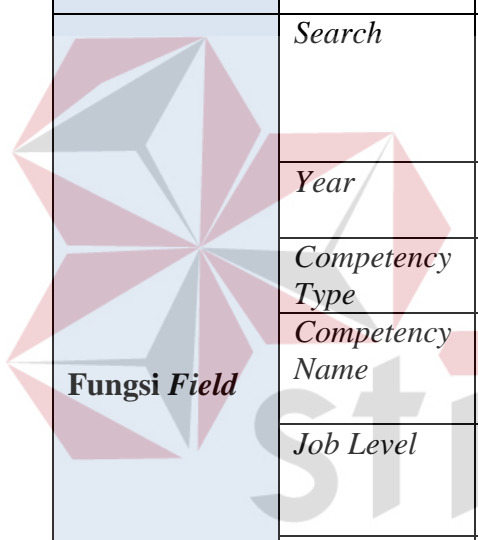
dalam penentuan kebutuhan pelatihan karyawan. Selain itu, pada kebijakan perusahaan, *section head* juga dapat melakukan penilaian dalam menentukan kompetensi karyawan. Namun, *section head* hanya dapat melakukan penilaian pada karyawan yang terdapat di bagiannya. Untuk mengetahui lebih jelasnya mengenai *form employee assessment* dapat dilihat pada Gambar 4.4.

Gambar 4.4 *Form Employee Assessment*

Adapun penjelasan lebih detail mengenai *form employee assessment* dan hasil penyesuaian antara *form* dengan rancangan yang dibuat sebelumnya dapat dilihat pada Tabel 4.3.

Tabel 4.3 Penjelasan *Form Employee Assessment*

Fungsional	Merencanakan Pengembangan Karyawan	
Deskripsi	<i>Form</i> ini digunakan untuk melakukan penentuan tingkat kompetensi karyawan dan penentuan kebutuhan pelatihan karyawan.	
Fungsi Tombol	<i>Save</i>	Tombol ini digunakan untuk menyimpan hasil penilaian kompetensi karyawan termasuk

		kesenjangan (<i>gap</i>) antara tingkat kompetensi karyawan dengan tingkat kompetensi jabatan. Selain itu, juga menyimpan data mengenai program pelatihan yang dibutuhkan oleh karyawan.
	<i>Competency Desc</i>	Berfungsi untuk melihat deskripsi singkat mengenai kompetensi yang akan digunakan dalam melakukan penilaian kompetensi karyawan.
	<i>Behavior Indicator Assessment</i>	Tombol ini akan digunakan untuk masuk ke dalam <i>Form Behavior Indicator Assessment</i> dalam melakukan proses penilaian indikator perilaku pada setiap tingkat kompetensi.
	<i>Close</i>	Berfungsi untuk keluar dari <i>Form Employee Assessment</i> beserta membatalkan proses penilaian kompetensi karyawan.
	<i>Search</i>	Digunakan untuk melakukan pencarian karyawan yang akan dinilai kompetensinya. Hasil pencarian akan muncul pada tabel yang terdapat pada <i>form</i> ini.
	<i>Year</i>	Berfungsi untuk memasukkan tahun pada penilaian kompetensi karyawan.
	<i>Competency Type</i>	Berfungsi untuk menentukan jenis kompetensi pada karyawan yang akan dinilai.
	<i>Competency Name</i>	Berfungsi untuk menentukan nama kompetensi pada karyawan yang akan dinilai sesuai dengan jenis kompetensi yang dipilih.
	<i>Job Level</i>	Berfungsi untuk menampilkan tingkat kompetensi yang dibutuhkan oleh jabatan dalam melakukan pekerjaan.
	<i>Employee Level</i>	Berfungsi untuk menampilkan tingkat kompetensi karyawan setelah dilakukan penilaian pada indikator perilaku.
	<i>GAP</i>	Berfungsi untuk menampilkan kesenjangan (<i>gap</i>) antara antara tingkat kompetensi karyawan dengan tingkat kompetensi jabatan.

Untuk melakukan penilaian terhadap indikator perilaku yang terdapat pada setiap tingkat kompetensi, aplikasi ini telah menyediakan *form behavior indicator assessment*. Untuk mengetahui *form* tersebut dapat melihatnya pada Gambar 4.5.

Dalam melakukan penilaian pada setiap indikator perilaku, pengguna dapat melakukan dengan memilih salah satu indikator perilaku yang terdapat pada tabel. Kemudian aplikasi akan memunculkan *form level competent* untuk menentukan apakah karyawan telah berkompeten atau tidak dalam indikator perilaku tersebut. *form level competent* dapat dilihat pada Gambar 4.6.

Indicator	Level Competent
Berinteraksi dengan baik kepada semua pelanggan	Competent
Memahami semua layanan yang akan diberikan kepada pelanggan	Quite Competent
Memperlakukan semua pelanggan dengan adil dan merata	Competent
Mendengarkan dan merespon kebutuhan pelanggan dengan memberikan informasi atau saran yang benar dan relevan dengan mempertimba...	Competent
Mengikuti permintaan pelanggan sampai akhir (jika memungkinkan) untuk memastikan kebutuhan pelanggan terpenuhi	Competent

Gambar 4.5 *Form Behavior Indicator Assessment*

Level Competent

Indicator Name : Mengikuti permintaan pelanggan sampai akhir (jika memungkinkan) untuk memastikan kebutuhan pelanggan terpenuhi

Not Competent
 Quite Competent
 Competent

Gambar 4.6 *Form Level Competent*

Adapun penjelasan lebih detail mengenai *Form Behavior Indicator Assessment* beserta hasil penyesuaian antara *form* dengan rancangan yang dibuat sebelumnya dapat dilihat pada Tabel 4.4. Penjelasan mengenai *Form Level Competent* juga akan dijelaskan pada Tabel 4.5. *Form Level Competent* tersebut akan digunakan untuk menilai tingkat kompeten pada setiap indikator perilaku.

Tabel 4.4 Penjelasan *Form Behavior Indicator Assessment*

Fungsional	Merencanakan Pengembangan Karyawan	
Deskripsi	Pada <i>form</i> ini digunakan untuk melakukan penentuan tingkat kompetensi karyawan berdasarkan hasil penilaian indikator perilaku. Penentuan tersebut akan dilakukan oleh sistem dengan perhitungan yang telah disediakan sebelumnya.	
Fungsi Tombol	<i>Next</i>	Berfungsi untuk melanjutkan penilaian indikator perilaku pada tingkat kompetensi selanjutnya.
	<i>Prev</i>	Berfungsi untuk melihat kembali nilai indikator perilaku pada tingkat kompetensi sebelumnya.
	<i>Finish</i>	Berfungsi untuk menyelesaikan penilaian indikator perilaku yang dilakukan oleh karyawan. Tombol ini akan aktif jika sistem telah menentukan tingkat kompetensi karyawan berdasarkan hasil dari penilaian indikator perilaku.
	<i>Close</i>	Berfungsi untuk kembali ke <i>Form Employee Assessment</i> dan membatalkan penilaian indikator perilaku yang dilakukan oleh karyawan.

Tabel 4.5 Penjelasan *Form Level Competent*

Fungsional	Merencanakan Pengembangan Karyawan	
Deskripsi	<i>Form</i> ini digunakan oleh <i>department head</i> dan <i>section head</i> untuk menentukan atau menilai tingkat kompeten yang dari indikator perilaku yang dimiliki oleh karyawan.	
Fungsi Tombol	<i>Icon centang</i>	Berfungsi untuk menyelesaikan penentuan tingkat kompeten karyawan terhadap indikator perilaku yang dinilai. Jika karyawan telah berkompeten dalam melaksanakan salah satu indikator perilaku, maka penilai memilih " <i>competent</i> " pada <i>radiobutton</i> yang disediakan. Jika cukup berkompeten maka penilai memilih " <i>Quite Competent</i> " dan jika tidak berkompeten maka penilai memilih " <i>Not Competent</i> ".
	<i>Icon silang</i>	Berfungsi untuk kembali ke <i>form behavior indicator assessment</i> dan membatalkan proses dalam menentukan tingkat kompeten karyawan terhadap indikator perilaku.

Untuk melihat dan mencetak hasil penilaian kompetensi karyawan, aplikasi ini menyediakan *form assessment report*. Hasil dari penilaian tersebut akan diberikan kepada karyawan yang telah dinilai sebagai laporan penilaian

kompetensi. Untuk mengetahui *form assessment report* dapat dilihat pada Gambar 4.7.

Gambar 4.7 *Form Assessment Report*

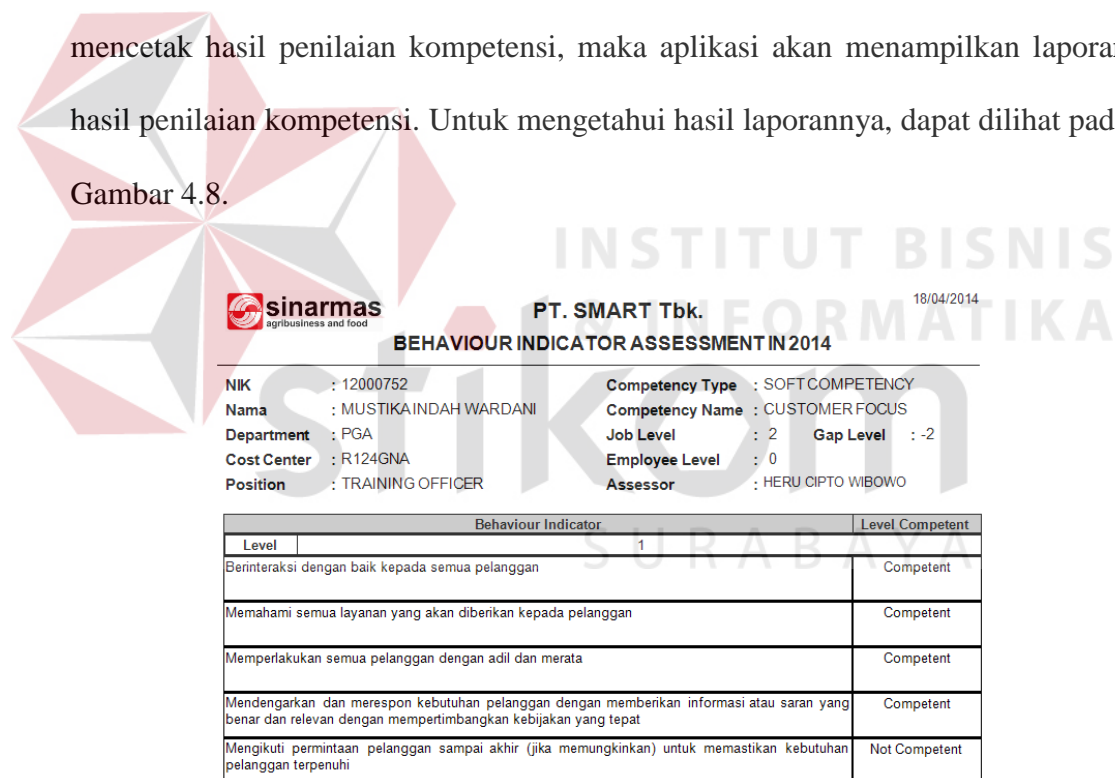
Adapun penjelasan lebih detail mengenai *form assessment report* beserta hasil penyesuaian antara *form* dengan rancangan yang dibuat sebelumnya dapat dilihat pada Tabel 4.6.

Tabel 4.6 Penjelasan *Form Assessment Report*

Fungsional	Merencanakan Pengembangan Karyawan	
Deskripsi	Pada <i>form</i> ini digunakan untuk membuat dan menampilkan laporan hasil penilaian kompetensi karyawan pada setiap tahun.	
Fungsi Tombol	<i>View Assessment Report</i>	Berfungsi untuk menampilkan laporan hasil penilaian kompetensi setiap karyawan sesuai dengan nama karyawan dan periode tahun yang telah dipilih atau ditentukan.
	<i>Close</i>	Berfungsi untuk keluar dari <i>form assessment report</i> dan membatalkan untuk melihat laporan hasil penilaian kompetensi setiap karyawan.
Fungsi Field	<i>Department</i>	Digunakan untuk memilih nama departemen.
	<i>Cost Center</i>	Digunakan dalam memilih <i>Cost Center</i> sesuai dengan nama departemen yang telah dipilih.
	<i>Position</i>	Digunakan untuk memilih nama jabatan sesuai dengan <i>Cost Center</i> dan nama departemen yang dipilih.

	<i>Superior</i>	Digunakan untuk menampilkan atasan pada nama jabatan yang telah dipilih
	<i>Employee Name</i>	Digunakan untuk memilih nama karyawan yang akan dilihat laporan hasil penilaian kompetensi setiap karyawan.
	<i>Competency Type</i>	Digunakan untuk memilih jenis kompetensi.
	<i>Competency Name</i>	Digunakan untuk memilih nama kompetensi sesuai dengan jenis kompetensi yang telah dipilih.
	<i>Year</i>	Digunakan untuk menentukan periode tahun pada laporan hasil penilaian kompetensi setiap karyawan.

Setelah memilih nama karyawan dan nama kompetensi untuk melihat dan mencetak hasil penilaian kompetensi, maka aplikasi akan menampilkan laporan hasil penilaian kompetensi. Untuk mengetahui hasil laporannya, dapat dilihat pada Gambar 4.8.



Behaviour Indicator		Level Competent
Level	1	
Berinteraksi dengan baik kepada semua pelanggan		Competent
Memahami semua layanan yang akan diberikan kepada pelanggan		Competent
Memperlakukan semua pelanggan dengan adil dan merata		Competent
Mendengarkan dan merespon kebutuhan pelanggan dengan memberikan informasi atau saran yang benar dan relevan dengan mempertimbangkan kebijakan yang tepat		Competent
Mengikuti permintaan pelanggan sampai akhir (jika memungkinkan) untuk memastikan kebutuhan pelanggan terpenuhi		Not Competent

Gambar 4.8 Laporan Hasil Penilaian Kompetensi Karyawan

Dalam membuat laporan kebutuhan pelatihan, aplikasi ini menyediakan *form department analysis* dan *form individual analysis* yang digunakan untuk melihat laporan analisis kebutuhan pelatihan pada setiap departemen atau semua

departemen di dalam perusahaan. Untuk mengetahui lebih jelasnya mengenai *form department analysis* dapat dilihat pada Gambar 4.9.


Gambar 4.9 *Form Department Analysis*

Adapun penjelasan lebih detail mengenai *form department analysis* beserta hasil penyesuaian antara *form* dengan rancangan yang dibuat sebelumnya dapat dilihat pada Tabel 4.7.

Tabel 4.7 *Penjelasan Form Department Analysis*

Fungsional	Merencanakan Pengembangan Karyawan	
Deskripsi	Pada <i>form</i> ini digunakan untuk membuat dan menampilkan laporan kebutuhan pelatihan karyawan pada setiap tahun.	
Fungsi Tombol	<i>View Report</i>	Berfungsi untuk menampilkan laporan kebutuhan pelatihan karyawan sesuai dengan kategori laporan yang telah dipilih atau ditentukan.
	<i>Close</i>	Berfungsi untuk keluar dari <i>form department analysis</i> dan membatalkan untuk melihat laporan kebutuhan pelatihan karyawan.
Fungsi Field	<i>Department</i>	Digunakan untuk memilih kategori laporan berdasarkan nama departemen atau semua departemen.
	<i>Competency Type</i>	Digunakan dalam memilih kategori laporan berdasarkan jenis kompetensi jika pengguna tidak memilih kategori laporan untuk semua kompetensi.
	<i>Competency Name</i>	Digunakan untuk memilih nama kompetensi sesuai dengan jenis kompetensi yang dipilih.
	<i>Year</i>	Digunakan untuk menentukan kategori laporan berdasarkan periode tahun.

Setelah memilih kategori laporan yang akan dicetak, maka aplikasi akan menampilkan hasil laporan sesuai dengan kategori yang dipilih. Apabila pengguna memilih kategori laporan berdasarkan nama departemen dan nama kompetensi, maka hasil laporannya dapat dilihat pada Gambar 4.10.

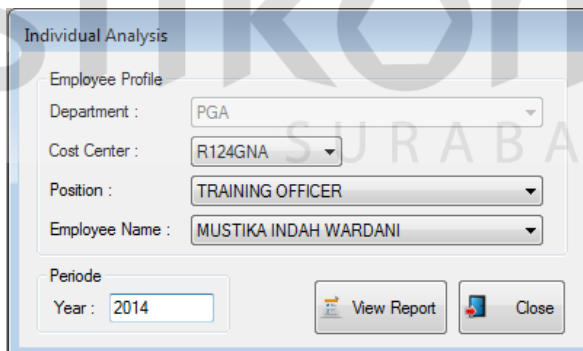
 **PT. SMART Tbk.** 17/04/2014

TRAINING NEED ANALYSIS REPORT IN 2014

NIK	Name	Department	Cost Center	Position
5010768	KARMELIA JULIANI	PGA	R124GNA	RECRUITMENT OFFICER
SOFT COMPETENCY		CUSTOMER FOCUS	BASIC CUSTOMER SERVICE EXCELLENCE PROGRAMME	
SOFT COMPETENCY		CUSTOMER FOCUS	ADVANCED CUSTOMER SERVICE EXCELLENCE PROGRAMME	
12000752	MUSTIKA INDAH WARDANI	PGA	R124GNA	TRAINING OFFICER
SOFT COMPETENCY		CUSTOMER FOCUS	BASIC CUSTOMER SERVICE EXCELLENCE PROGRAMME	
SOFT COMPETENCY		CUSTOMER FOCUS	ADVANCED CUSTOMER SERVICE EXCELLENCE PROGRAMME	

Gambar 4.10 Laporan Kebutuhan Pelatihan Karyawan

Untuk mengetahui kebutuhan pelatihan setiap individu atau karyawan, dapat dilihat pada *form individual analysis*. *Form* tersebut akan digambarkan pada Gambar 4.11.



Individual Analysis

Employee Profile

Department : PGA

Cost Center : R124GNA

Position : TRAINING OFFICER

Employee Name : MUSTIKA INDAH WARDANI

Periode

Year : 2014

View Report Close

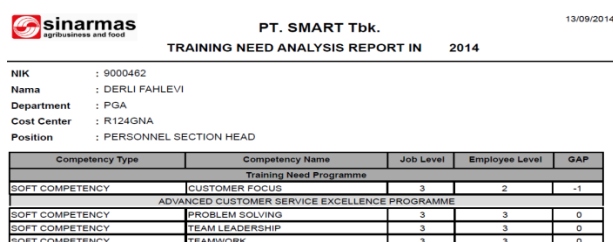
Gambar 4.11 *Form Individual Analysis*

Adapun penjelasan lebih detail mengenai *form individual analysis* beserta hasil penyesuaian antara *form* dengan rancangan yang dibuat sebelumnya dapat dilihat pada Tabel 4.8.

Tabel 4.8 Penjelasan *Form Individual Analysis*

Fungsional	Merencanakan Pengembangan Karyawan	
Deskripsi	Pada <i>form</i> ini digunakan untuk membuat dan menampilkan laporan kebutuhan pelatihan setiap individu atau karyawan pada setiap tahun.	
Fungsi Tombol	<i>View Report</i>	Berfungsi untuk menampilkan laporan kebutuhan pelatihan setiap karyawan sesuai dengan nama karyawan dan periode tahun yang telah dipilih atau ditentukan.
	<i>Close</i>	Berfungsi untuk keluar dari <i>form individual analysis</i> dan membatalkan untuk melihat laporan kebutuhan pelatihan setiap karyawan.
Fungsi Field	<i>Department</i>	Digunakan untuk memilih nama departemen.
	<i>Cost Center</i>	Digunakan dalam memilih <i>Cost Center</i> sesuai dengan nama departemen yang telah dipilih.
	<i>Position</i>	Digunakan untuk memilih nama jabatan sesuai dengan <i>Cost Center</i> dan nama departemen yang dipilih.
	<i>Employee Name</i>	Digunakan untuk memilih nama karyawan yang akan dilihat laporan kebutuhannya.
	<i>Year</i>	Digunakan untuk menentukan periode tahun pada laporan kebutuhan pelatihan setiap karyawan.

Setelah memilih nama karyawan yang akan dicetak laporan kebutuhan pelatihannya, maka aplikasi akan menampilkan hasil laporan sesuai dengan nama karyawan yang dipilih. Untuk mengetahui hasil laporannya dapat dilihat pada Gambar 4.12.



Competency Type	Competency Name	Job Level	Employee Level	GAP
Training Need Programme				
SOFT COMPETENCY	CUSTOMER FOCUS	3	2	-1
ADVANCED CUSTOMER SERVICE EXCELLENCE PROGRAMME				
SOFT COMPETENCY	PROBLEM SOLVING	3	3	0
SOFT COMPETENCY	TEAM LEADERSHIP	3	3	0
SOFT COMPETENCY	TEAMWORK	3	3	0

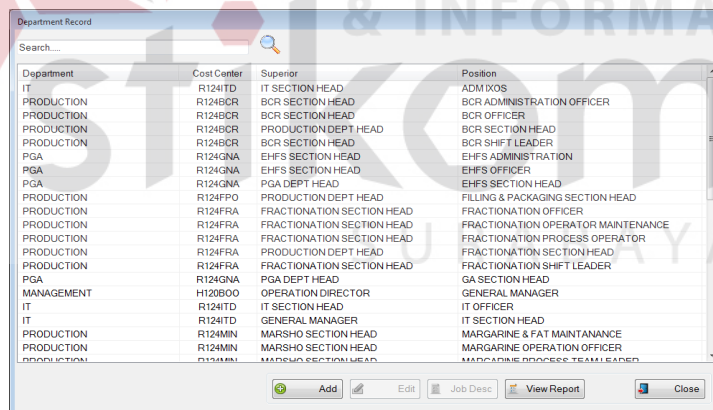
Gambar 4.12 Laporan Kebutuhan Pelatihan Setiap Karyawan

4.3.2 Pengguna Sebagai *Training Officer*

Training officer melakukan 3 fungsional dalam sistem, yaitu:

1. Melakukan *Update Active File*

Fungsional melakukan *update active file* memiliki beberapa *form* yang digunakan untuk mengelola data induk atau *active file*, yaitu data jabatan, data karyawan dan data kompetensi. Data-data tersebut merupakan akan digunakan dalam melaksanakan proses analisis kebutuhan pelatihan karyawan. Sebelum melakukan *input* data karyawan, diharapkan untuk melakukan *input* data jabatan terlebih dahulu. Hal tersebut dikarenakan data karyawan tergantung terhadap adanya data jabatan yang tersedia. Untuk mengetahui lebih jelasnya mengenai *input* data jabatan beserta pengelolaannya dapat dilihat *form department record* pada Gambar 4.13.



Department	Cost Center	Superior	Position
IT	R124ITD	IT SECTION HEAD	ADM IXOS
PRODUCTION	R124BCR	BCR SECTION HEAD	BCR ADMINISTRATION OFFICER
PRODUCTION	R124BCR	BCR SECTION HEAD	BCR OFFICER
PRODUCTION	R124BCR	PRODUCTION DEPT HEAD	BCR SECTION HEAD
PRODUCTION	R124BCR	BCR SECTION HEAD	BCR SHIFT LEADER
PGA	R124GNA	EHFS SECTION HEAD	EHFS ADMINISTRATION
PGA	R124GNA	EHFS SECTION HEAD	EHFS OFFICER
PGA	R124GNA	PGA DEPT HEAD	EHFS SECTION HEAD
PRODUCTION	R124FPO	PRODUCTION DEPT HEAD	FILLING & PACKAGING SECTION HEAD
PRODUCTION	R124FRA	FRACTIONATION SECTION HEAD	FRACTIONATION OFFICER
PRODUCTION	R124FRA	FRACTIONATION SECTION HEAD	FRACTIONATION OPERATOR MAINTENANCE
PRODUCTION	R124FRA	FRACTIONATION SECTION HEAD	FRACTIONATION PROCESS OPERATOR
PRODUCTION	R124FRA	PRODUCTION DEPT HEAD	FRACTIONATION SECTION HEAD
PRODUCTION	R124FRA	FRACTIONATION SECTION HEAD	FRACTIONATION SHIFT LEADER
PGA	R124GNA	PGA DEPT HEAD	GA SECTION HEAD
MANAGEMENT	H120B00	OPERATION DIRECTOR	GENERAL MANAGER
IT	R124ITD	IT SECTION HEAD	IT OFFICER
IT	R124ITD	GENERAL MANAGER	IT SECTION HEAD
PRODUCTION	R124MIN	MARSHO SECTION HEAD	MARGARINE & FAT MAINTANANCE
PRODUCTION	R124MIN	MARSHO SECTION HEAD	MARGARINE OPERATION OFFICER
PRODUCTION	R124MIN	MARSHO SECTION HEAD	MARGARINE PROCESS TEAM LEADER

Gambar 4.13 *Form Department Record*

Adapun penjelasan lebih detil mengenai *form department record* beserta hasil penyesuaian antara *form* dengan rancangan yang dibuat sebelumnya dapat dilihat pada Tabel 4.9.

Tabel 4.9 Penjelasan *Form Department Record*

Fungsional	Melakukan <i>Update Active File</i>	
Deskripsi	Pada <i>form</i> ini digunakan untuk melakukan pengelolaan data jabatan di dalam sebuah departemen.	
Fungsi Tombol	<i>Add</i>	Berfungsi untuk menambahkan jabatan baru di dalam departemen.
	<i>Edit</i>	Berfungsi untuk mengganti nama jabatan yang dipilih di dalam departemen.
	<i>Job Desc</i>	Berfungsi untuk melihat <i>Job Description</i> dari nama jabatan yang dipilih. <i>Job Description</i> tersebut berupa dokumen pdf yang berisi mengenai deskripsi singkat pada jabatan.
	<i>View Report</i>	Berfungsi untuk menampilkan sebuah laporan data jabatan yang tersedia di setiap departemen.
	<i>Close</i>	Berfungsi untuk keluar dari <i>form department record</i> .
Fungsi Field	<i>Search</i>	Digunakan untuk melakukan pencarian data jabatan. Hasil pencarian akan muncul pada tabel yang terdapat pada <i>form</i> ini.

Berikut ini merupakan tampilan dari *form add/edit position* untuk menambahkan dan merubah data jabatan. Untuk lebih jelasnya dapat dilihat pada Gambar 4.14.

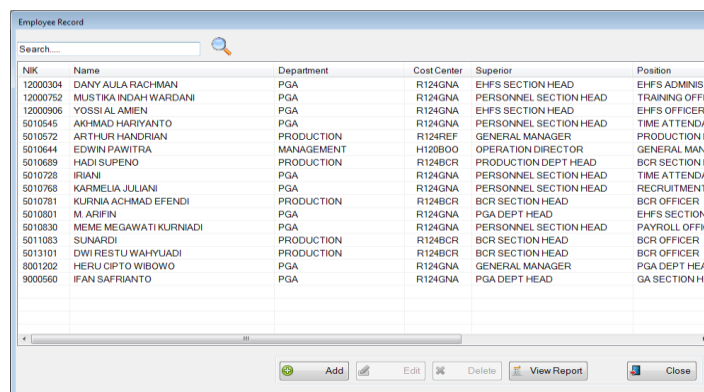
Gambar 4.14 *Form Add/Edit Position*

Adapun penjelasan lebih detil mengenai *form add/edit position* beserta hasil penyesuaian antara *form* dengan rancangan yang dibuat sebelumnya dapat dilihat pada Tabel 4.10.

Tabel 4.10 Penjelasan *Form Add/Edit Position*

Fungsional	Melakukan <i>Update Active File</i>	
Deskripsi	Pada <i>form</i> ini digunakan untuk melakukan penambahan atau penggantian nama jabatan di dalam suatu departemen.	
Fungsi Tombol	<i>Save</i>	Berfungsi untuk menyimpan data jabatan ke dalam <i>database</i> setelah semua <i>field</i> dari <i>form</i> ini terisi.
	<i>Back</i>	Berfungsi untuk kembali ke <i>form department record</i> dan membatalkan penambahan atau penggantian nama jabatan.
	<i>Browse</i>	Berfungsi untuk mengambil lokasi tempat <i>file Job Description</i> di komputer pengguna.
Fungsi Field	<i>Department Name</i>	Digunakan untuk mengisikan nama departemen atau memilih nama departemen yang sudah pernah diisikan di dalam <i>database</i> .
	<i>Cost Center</i>	Digunakan untuk mengisikan <i>Cost Center</i> atau memilih <i>Cost Center</i> yang sudah pernah diisikan di dalam <i>database</i> .
	<i>Superior</i>	Digunakan untuk mengisikan nama atasan pada sebuah jabatan atau memilih nama atasan yang sudah pernah diisikan di dalam <i>database</i> .
	<i>Position</i>	Digunakan untuk mengisikan nama jabatan.
	<i>File Attachment</i>	Berfungsi untuk menampilkan lokasi tempat <i>file Job Description</i> setelah dipilih di computer pengguna.

Untuk melakukan pengelolaan data karyawan, aplikasi ini telah menyediakan *form employee record*. Berikut ini merupakan tampilan dari *form employee record* yang dapat dilihat pada Gambar 4.15.



NIK	Name	Department	Cost Center	Superior	Position
12000304	DANNY AJULA RACHMAN	PGA	R124GNA	EHFS SECTION HEAD	EHFS ADMINIST
12000752	MUSTIKA INDAH WARDANI	PGA	R124GNA	PERSONNEL SECTION HEAD	TRAINING OFFIC
12000906	YOSSI AL AMIEN	PGA	R124GNA	EHFS SECTION HEAD	EHFS OFFICER
5010545	AKHMAD HARIYANTO	PGA	R124GNA	PERSONNEL SECTION HEAD	TIME ATTENDAY
5010572	ARTHUR HANDRIAN	PRODUCTION	R124REF	GENERAL MANAGER	PRODUCTION D
5010644	EDWIN PANITRA	MANAGEMENT	H128900	OPERATION DIRECTOR	GENERAL MANU
5010659	HADI SUPENO	PRODUCTION	R124BCR	PRODUCTION DEPT HEAD	BCR SECTION H
5010728	IRIANI	PGA	R124GNA	PERSONNEL SECTION HEAD	TIME ATTENDAY
5010768	KARMELIA JULIANI	PGA	R124GNA	PERSONNEL SECTION HEAD	RECRUITMENT
5010781	KURNIA ACHMAD EFENDI	PRODUCTION	R124BCR	BCR SECTION HEAD	BCR OFFICER
5010801	M. ARIFIN	PGA	R124GNA	PGA DEPT HEAD	EHFS SECTION
5010830	MEME MEGAWATI KURNIADI	PGA	R124GNA	PERSONNEL SECTION HEAD	PAYROLL OFFIC
5011083	SUNARDI	PRODUCTION	R124BCR	BCR SECTION HEAD	BCR OFFICER
5013101	DWI RESTU WAHYUADI	PRODUCTION	R124BCR	BCR SECTION HEAD	BCR OFFICER
8001202	HERU CIPTO WIBOWO	PGA	R124GNA	GENERAL MANAGER	PGA DEPT HEAT
9000560	IFAN SAFFRIANTO	PGA	R124GNA	PGA DEPT HEAD	GA SECTION HE

Gambar 4.15 *Form Employee Record*

Adapun penjelasan lebih detail mengenai *form employee record* beserta hasil penyesuaian antara *form* dengan rancangan yang dibuat sebelumnya dapat dilihat pada Tabel 4.11.

Tabel 4.11 Penjelasan *Form Employee Record*

Fungsional	Melakukan <i>Update Active File</i>	
Deskripsi	Pada <i>form</i> ini digunakan untuk melakukan pengelolaan data karyawan yang terdapat pada perusahaan.	
Fungsi Tombol	<i>Add</i>	Berfungsi untuk menambahkan data karyawan baru yang memulai kerja di perusahaan.
	<i>Edit</i>	Berfungsi untuk mengganti data karyawan yang terdapat di perusahaan.
	<i>Delete</i>	Berfungsi untuk menghapus data karyawan yang keluar atau tidak bekerja di perusahaan.
	<i>View Report</i>	Berfungsi untuk menampilkan laporan data karyawan yang terdapat di perusahaan.
	<i>Close</i>	Berfungsi untuk keluar dari <i>form employee record</i> .
Fungsi Field	<i>Search</i>	Digunakan untuk melakukan pencarian data karyawan. Hasil pencarian akan muncul pada tabel yang terdapat pada <i>form</i> ini.

Berikut ini merupakan tampilan dari *form add/edit employee* untuk menambahkan dan merubah data karyawan. Untuk lebih jelasnya dapat dilihat pada Gambar 4.16.

Gambar 4.16 *Form Add/Edit Employee*

Adapun penjelasan lebih detail mengenai *form add/edit employee* beserta hasil penyesuaian antara *form* dengan rancangan yang dibuat sebelumnya dapat dilihat pada Tabel 4.12.

Tabel 4.12 Penjelasan *Form Add/Edit Employee*

Fungsional	Melakukan <i>Update Active File</i>	
Deskripsi	Pada <i>form</i> ini digunakan untuk melakukan penambahan atau penggantian data karyawan yang terdapat pada perusahaan.	
Fungsi Tombol	<i>Save</i>	Berfungsi untuk menyimpan data karyawan ke dalam <i>database</i> setelah semua <i>field</i> dari <i>form</i> ini terisi.
	<i>Back</i>	Berfungsi untuk kembali ke <i>form employee record</i> dan membatalkan penambahan atau penggantian data karyawan.
	<i>Browse</i>	Berfungsi untuk mengambil lokasi tempat <i>file</i> foto karyawan di komputer pengguna.
Fungsi Field	<i>NIK</i>	Digunakan untuk mengisikan nomer induk karyawan pada perusahaan.
	<i>Name</i>	Digunakan untuk mengisikan nama karyawan yang bekerja pada perusahaan.
	<i>Department Name</i>	Digunakan untuk memilih nama departemen yang terdapat di dalam <i>database</i> .
	<i>Cost Center</i>	Digunakan memilih <i>Cost Center</i> yang terdapat di dalam <i>database</i> .
	<i>Superior</i>	Digunakan untuk menampilkan nama atasan sesuai dengan nama jabatan dan departemen yang dipilih.
	<i>Position</i>	Digunakan untuk memilih nama jabatan yang terdapat di dalam <i>database</i> sesuai dengan departemen yang dipilih.
	<i>Group</i>	Digunakan untuk mengisikan golongan karyawan.
	<i>Entry Date</i>	Digunakan untuk mengisikan tanggal pada saat karyawan mulai bekerja di perusahaan.
	<i>Date of Birth</i>	Digunakan untuk mengisikan tanggal lahir karyawan.
	<i>Status</i>	Digunakan untuk memilih status karyawan. Status karyawan ada dua jenis, yaitu staff dan non staff.
	<i>File Photo</i>	Berfungsi untuk menampilkan lokasi tempat <i>file</i> foto karyawan setelah dipilih di computer pengguna.

Untuk melakukan pengelolaan data kompetensi, aplikasi ini telah menyediakan *form competency record*. Berikut ini merupakan tampilan dari *form competency record* yang dapat dilihat pada Gambar 4.17.

Gambar 4.17 *Form Competency Record*

Adapun penjelasan lebih detail mengenai *form competency record* beserta hasil penyesuaian antara *form* dengan rancangan yang dibuat sebelumnya dapat dilihat pada Tabel 4.13.

Tabel 4.13 Penjelasan *Form Competency Record*

Fungsional	Melakukan <i>Update Active File</i>	
Deskripsi	Pada <i>form</i> ini digunakan untuk melakukan pengelolaan data kompetensi yang harus dimiliki oleh karyawan di perusahaan.	
Fungsi Tombol	<i>Add</i> "Competency"	Berfungsi untuk menambahkan data kompetensi yang harus dimiliki oleh karyawan di perusahaan.
	<i>Edit</i> "Competency"	Berfungsi untuk mengganti data kompetensi yang harus dimiliki oleh karyawan di perusahaan.
	<i>View Desc</i>	Berfungsi untuk melihat dokumen yang berisi deskripsi singkat mengenai data kompetensi yang dipilih.
	<i>Add</i> "Behavior Indicators"	Berfungsi untuk menambahkan data indikator perilaku pada setiap tingkat kompetensi.

	<i>Edit “Behavior Indicators”</i>	Berfungsi untuk mengganti data indikator perilaku pada setiap tingkat kompetensi.
	<i>Close</i>	Berfungsi untuk keluar dari <i>form competency record</i> .
Fungsi Field	<i>Search</i>	Digunakan untuk melakukan pencarian data kompetensi. Hasil pencarian akan muncul pada tabel yang terdapat pada <i>form</i> ini di bagian <i>competency</i> .
	<i>Level</i>	Digunakan untuk memilih tingkat kompetensi dan kemudian tabel pada <i>form</i> ini akan menampilkan indikator perilaku berdasarkan tingkat kompetensi yang dipilih.

Berikut ini merupakan tampilan dari *form add/edit competency* untuk menambahkan dan mengganti data kompetensi yang harus dimiliki oleh karyawan di perusahaan. Untuk lebih jelasnya dapat dilihat pada Gambar 4.18.

Gambar 4.18 *Form Add/Edit Competency*

Adapun penjelasan lebih detail mengenai *form add/edit competency* beserta hasil penyesuaian antara *form* dengan rancangan yang dibuat sebelumnya dapat dilihat pada Tabel 4.14.

Tabel 4.14 Penjelasan *Form Add/Edit Competency*

Fungsional	Melakukan <i>Update Active File</i>	
Deskripsi	Pada <i>form</i> ini digunakan untuk melakukan penambahan atau penggantian data kompetensi yang harus dimiliki oleh karyawan di perusahaan.	
Fungsi Tombol	<i>Save</i>	Berfungsi untuk menyimpan data kompetensi ke dalam <i>database</i> setelah semua <i>field</i> dari <i>form</i> ini terisi.

	<i>Back</i>	Berfungsi untuk kembali ke <i>form add/edit competency</i> dan membatalkan penambahan atau penggantian data kompetensi.
	<i>Browse</i>	Berfungsi untuk mengambil lokasi tempat <i>file</i> kamus kompetensi yang mengenai deskripsi singkat kompetensi yang dibutuhkan perusahaan.
Fungsi Field	<i>Competency Type</i>	Digunakan untuk memilih jenis kompetensi. Jenis kompetensi terdiri dari dua, yaitu <i>soft competency</i> dan <i>hard competency</i> .
	<i>Competency Name</i>	Digunakan untuk mengisikan nama kompetensi yang dibutuhkan oleh perusahaan.
	<i>File Attachment</i>	Berfungsi untuk menampilkan lokasi tempat <i>file</i> kamus kompetensi yang mengenai deskripsi singkat tentang kompetensi.

Berikut ini merupakan tampilan dari *form add/edit behavior indicator* untuk menambahkan dan merubah data indikator perilaku pada setiap tingkat kompetensi. Tingkat kompetensi tersebut telah ditetapkan mulai dari tingkat 1 sampai dengan 4. Untuk mengetahui lebih jelasnya tentang *form add/edit behavior indicator* dapat dilihat pada Gambar 4.19.

Gambar 4.19 *Form Add/Edit Behavior Indicator*

Adapun penjelasan lebih detil mengenai *form add/edit behavior indicator* beserta hasil penyesuaian antara *form* dengan rancangan yang dibuat sebelumnya dapat dilihat pada Tabel 4.15.

Tabel 4.15 Penjelasan *Form Add/Edit Behavior Indicator*

Fungsional	Melakukan <i>Update Active File</i>	
Deskripsi	Pada <i>form</i> ini digunakan untuk melakukan penambahan atau penggantian data indikator perilaku yang terdapat pada setiap tingkat kompetensi.	
Fungsi Tombol	<i>Save</i>	Berfungsi untuk menyimpan data indikator perilaku ke dalam <i>database</i> setelah semua <i>field</i> dari <i>form</i> ini terisi.
	<i>Back</i>	Berfungsi untuk kembali ke <i>form add/edit behavior indicator</i> dan membatalkan penambahan atau penggantian data indikator perilaku pada setiap tingkat kompetensi.
Fungsi Field	<i>Competency Type</i>	Digunakan untuk menampilkan jenis kompetensi yang telah dipilih dalam menambahkan atau mengganti data indikator perilaku.
	<i>Competency Name</i>	Digunakan untuk menampilkan nama kompetensi yang telah dipilih dalam menambahkan atau merubah data indikator perilaku.
	<i>Level</i>	Digunakan untuk menampilkan tingkat kompetensi dalam menambahkan atau mengganti data indikator perilaku.
	<i>Training Program</i>	Digunakan untuk mengisi nama program pelatihan yang disediakan oleh perusahaan sesuai dengan peningkatan kompetensi yang dibutuhkan.
	<i>Indicator</i>	Digunakan untuk mengisi indikator perilaku yang terdapat pada setiap tingkat kompetensi.

2. Membuat Standar Kompetensi Jabatan

Fungsional ini terdiri dari 2 *form* yang digunakan untuk membuat standar kompetensi dalam setiap jabatan. *Form* tersebut adalah *form standard competency* dan *form add standard competency*. Standar kompetensi dapat ditambahkan jika data kompetensi dan data jabatan telah tersedia. Standar kompetensi tersebut merupakan sebuah standar kompetensi beserta tingkat kompetensi yang harus dimiliki dalam melakukan pekerjaan atau memegang jabatan. Untuk mengetahui lebih jelasnya mengenai *form standard competency* dapat dilihat pada Gambar 4.20.

Gambar 4.20 *Form Standard Competency*

Adapun penjelasan lebih detail mengenai *form standard competency* beserta hasil penyesuaian antara *form* dengan rancangan yang dibuat sebelumnya dapat dilihat pada Tabel 4.16.

Tabel 4.16 Penjelasan *Form Standard Competency*

Fungsional	Membuat Standar Kompetensi Jabatan	
Deskripsi	Pada <i>form</i> ini digunakan untuk melakukan pengelolaan data termasuk penambahan data standar kompetensi yang harus dimiliki oleh setiap jabatan.	
Fungsi Tombol	<i>Add</i>	Berfungsi untuk menambahkan data standar kompetensi yang dimiliki oleh setiap jabatan.
	<i>Job Desc</i>	Berfungsi untuk melihat <i>Job Description</i> dari jabatan yang telah dipilih.
	<i>Competency Desc</i>	Berfungsi untuk melihat deskripsi singkat mengenai kamus kompetensi yang telah dipilih dari tabel <i>form</i> ini.
	<i>View Report</i>	Berfungsi untuk menampilkan laporan data standar kompetensi yang dimiliki oleh setiap jabatan.
	<i>Close</i>	Berfungsi untuk keluar dari <i>form standard competency</i> .

Fungsi Field	<i>Department</i>	Digunakan untuk memilih nama departemen yang telah tersedia di dalam <i>database</i> .
	<i>Cost Center</i>	Digunakan untuk memilih <i>Cost Center</i> dari nama departemen yang telah dipilih
	<i>Position</i>	Digunakan untuk memilih nama jabatan dari nama departemen dan <i>Cost Center</i> yang telah dipilih. Setelah nama jabatan dipilih, <i>Superior</i> dari jabatan tersebut akan muncul dan data standar kompetensi dalam jabatan tersebut akan muncul di dalam tabel <i>form</i> ini.

Berikut ini merupakan tampilan dari *form add standard competency* untuk menambahkan data standar kompetensi yang harus dimiliki oleh setiap jabatan beserta tingkat kompetensi jabatannya. Untuk lebih jelasnya dapat dilihat pada Gambar 4.21.

Gambar 4.21 *Form Add Standard Competency*

Adapun penjelasan lebih detail mengenai *form add standard competency* beserta hasil penyesuaian antara *form* dengan rancangan yang dibuat sebelumnya dapat dilihat pada Tabel 4.17.

Tabel 4.17 Penjelasan *Form Add Standard Competency*

Fungsional	Membuat Standar Kompetensi Jabatan	
Deskripsi	Pada <i>form</i> ini digunakan untuk melakukan penambahan data standar kompetensi yang harus dimiliki oleh setiap jabatan beserta tingkat kompetensi yang dibutuhkan jabatan.	
Fungsi Tombol	<i>Save</i>	Berfungsi untuk melakukan penyimpanan data standar kompetensi yang dimiliki oleh setiap jabatan setelah semua <i>field</i> dari <i>form</i> ini terisi.
	<i>Job Desc</i>	Berfungsi untuk melihat <i>Job Description</i> dari jabatan yang telah dipilih.
	<i>Competency Desc</i>	Berfungsi untuk melihat deskripsi singkat mengenai kamus kompetensi yang telah dipilih dari <i>form</i> ini.
	<i>Back</i>	Berfungsi untuk kembali ke <i>form standard competency</i> dan membatalkan proses membuat standar kompetensi.
Fungsi Field	<i>Department</i>	Digunakan untuk menampilkan nama departemen yang telah dipilih sebelumnya pada <i>Form Standard Competency</i> .
	<i>Cost Center</i>	Digunakan untuk menampilkan <i>Cost Center</i> yang telah dipilih sebelumnya pada <i>form standard competency</i> .
	<i>Position</i>	Digunakan untuk menampilkan nama jabatan yang telah dipilih sebelumnya pada <i>form standard competency</i> .
	<i>Competency Type</i>	Digunakan untuk memilih jenis kompetensi yang telah tersedia.
	<i>Competency Name</i>	Digunakan untuk memilih nama kompetensi dari jenis kompetensi yang telah dipilih.
	<i>Job Level</i>	Digunakan untuk menentukan tingkat kompetensi yang dibutuhkan oleh jabatan.

3. Melaksanakan Perencanaan Pelatihan

Dalam fungsional melaksanakan perencanaan pelatihan, aplikasi ini menyediakan *form training participants* dan *form training need priority* yang digunakan untuk melihat laporan peserta yang akan mengikuti pelatihan pada setiap departemen atau semua departemen di dalam perusahaan beserta laporan prioritas pelaksanaan kebutuhan pelatihan. Untuk mengetahui lebih jelasnya mengenai *Form Training Participants* dapat dilihat pada Gambar 4.22.

Gambar 4.22 Form Training Participants

Adapun penjelasan lebih detail mengenai *form training participants* beserta hasil penyesuaian antara *form* dengan rancangan yang dibuat sebelumnya dapat dilihat pada Tabel 4.18.

Tabel 4.18 Penjelasan Form Training Participants

Fungsional	Melaksanakan Perencanaan Pelatihan	
Deskripsi	Pada <i>form</i> ini digunakan untuk membuat dan menampilkan laporan peserta yang akan mengikuti pelatihan pada setiap tahun.	
Fungsi Tombol	<i>View Report</i>	Berfungsi untuk menampilkan laporan peserta pelatihan sesuai dengan kategori laporan yang telah dipilih atau ditentukan.
	<i>Close</i>	Berfungsi untuk keluar dari <i>form department analysis</i> dan membatalkan untuk melihat laporan kebutuhan pelatihan karyawan.
Fungsi Field	<i>Department</i>	Digunakan untuk memilih kategori laporan berdasarkan nama departemen atau semua departemen.
	<i>Competency Type</i>	Digunakan dalam memilih kategori laporan berdasarkan jenis kompetensi jika pengguna tidak memilih kategori laporan untuk semua kompetensi.
	<i>Competency Name</i>	Digunakan untuk memilih nama kompetensi sesuai dengan jenis kompetensi yang dipilih.
	<i>Training Need Programme</i>	Digunakan untuk memilih nama program pelatihan yang akan diadakan sesuai dengan kebutuhan pelatihan.

	<i>Year</i>	Digunakan untuk menentukan kategori laporan berdasarkan periode tahun.
--	-------------	--

Setelah memilih kategori laporan yang akan dicetak, maka aplikasi akan menampilkan hasil laporan sesuai dengan kategori yang dipilih. Apabila pengguna memilih kategori laporan berdasarkan nama kompetensi beserta kebutuhan pelatihannya, maka hasil laporannya dapat dilihat pada Gambar 4.23.

sinarmas
agribusiness and food

PT. SMART Tbk. 10/09/2014

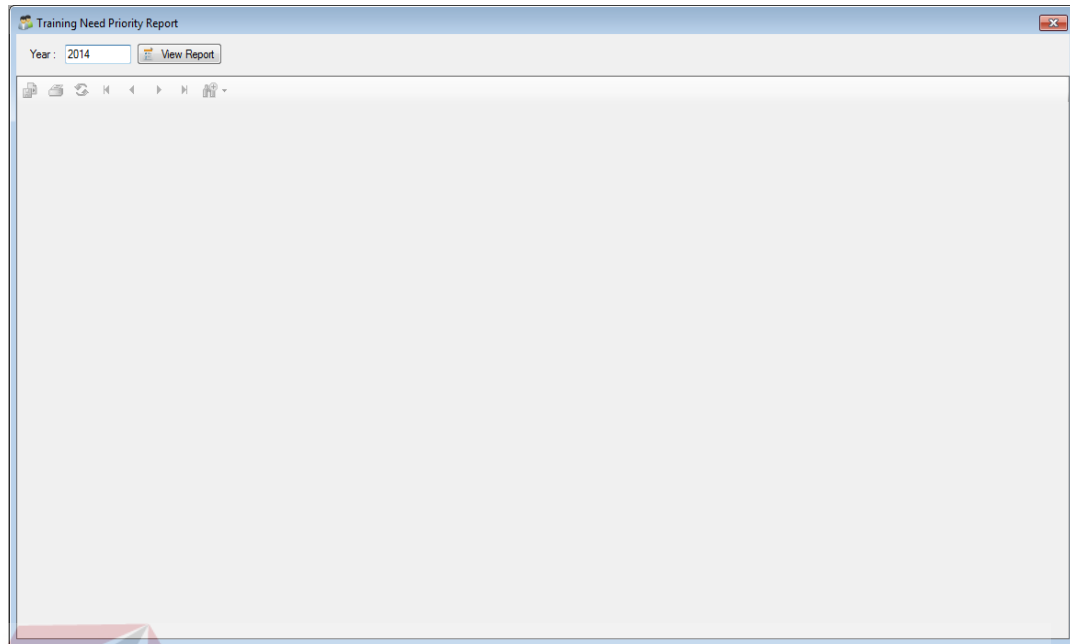
TRAINING PARTICIPANTS REPORT IN 2015

Competency Type : SOFT COMPETENCY
 Competency Name : CUSTOMER FOCUS
 Training Need Programme : ADVANCED CUSTOMER SERVICE EXCELLENCE PROGRAMME
 Number of Participants : 2

NIK	Nama	Position
9000462	DERLI FAHLEVI	PERSONNEL SECTION HEAD
9000560	IFAN SAFRIANTO	GA SECTION HEAD

Gambar 4.23 Laporan Peserta Pelatihan

Sedangkan jenis laporan lainnya adalah Laporan Prioritas Kebutuhan Pelatihan. Laporan tersebut digunakan untuk melihat prioritas pelaksanaan kebutuhan pelatihan yang akan diadakan. Untuk melihat dan mencetak laporan Laporan Prioritas Kebutuhan Pelatihan, maka aplikasi ini menyediakan *form training need priority report*. Lebih jelasnya dapat dilihat pada Gambar 4.24.



Gambar 4.24 *Form Training Need Priority Report*


Adapun penjelasan lebih detail mengenai *form training need priority report* beserta hasil penyesuaian antara *form* dengan rancangan yang dibuat sebelumnya dapat dilihat pada Tabel 4.19.

Tabel 4.19 Penjelasan *Form Training Need Priority Report*

Fungsional	Melaksanakan Perencanaan Pelatihan	
Deskripsi	Pada <i>form</i> ini digunakan untuk membuat dan menampilkan laporan prioritas kebutuhan pelatihan karyawan dalam pelaksanaannya.	
Fungsi Tombol	<i>View Report</i>	Berfungsi untuk menampilkan laporan prioritas kebutuhan pelatihan karyawan periode tahun yang telah ditentukan.
Fungsi Field	<i>Year</i>	Digunakan untuk menentukan periode tahun pada laporan prioritas kebutuhan pelatihan karyawan.

Setelah menentukan periode tahun dalam melaksanakan kebutuhan pelatihan, maka aplikasi akan menampilkan hasil laporan prioritas pelaksanaan kebutuhan

pelatihan sesuai dengan tahun yang ditentukan. Untuk mengetahui hasil laporannya dapat dilihat pada Gambar 4.23.



PT. SMART Tbk. 13/09/2014

PRIORITY TRAINING NEED REPORT IN 2015

Priority Number	Training Need Programme	Number of Participants	Average Gap	Status
1	BASIC CUSTOMER SERVICE EXCELLENCE PROGRAMME	2	-1,50	High Priority
2	BASIC PROBLEM SOLVING PROGRAMME	3	-1,30	Medium Priority
3	INTERMEDIATE CUSTOMER SERVICE EXCELLENCE PROGRAMME	3	-1,30	Medium Priority
4	INTERMEDIATE PROBLEM SOLVING PROGRAMME	5	-1,20	Medium Priority
5	INTERMEDIATE TEAMWORK PROGRAMME	1	-1,00	Low Priority

Gambar 4.25 Laporan Prioritas Kebutuhan Pelatihan

4.4 Uji Coba Fungsional dan Non-Fungsional


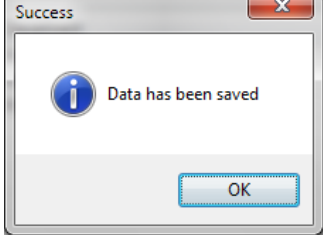
Pada tahap ini akan dilakukan uji coba terhadap perangkat lunak yang telah dibangun, apakah perangkat lunak tersebut sesuai dengan fungsional dan non-fungsional yang telah dirancang sebelumnya. Pada uji coba fungsional yang dilakukan, ada beberapa tampilan aplikasi yang sama pada saat implementasi, sehingga pada saat uji coba fungsional, penjelasan akan dihubungkan pada tahap implementasi. Dalam artian pada saat melakukan implementasi, penulis sekaligus melakukan uji coba fungsional.

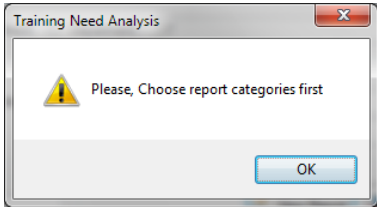
4.4.1 Uji Coba Pada *Department Head*

Mengenai uji fungsional merencanakan pengembangan karyawan dalam menentukan kebutuhan pelatihan lebih detil yang dilakukan oleh *Department Head* dapat dilihat pada Tabel 4.20.

Tabel 4.20 Hasil Uji Fungsional Merencanakan Pengembangan Karyawan

Fungsional	Merencanakan Pengembangan Karyawan	
Stakeholder	Department head	
Alur Normal		
Aksi Pengguna	Respon Sistem	Hasil
Pengguna memilih sub menu “ <i>Employee Assessment</i> ” pada menu “ <i>Assessment Process</i> ”.	Sistem akan menampilkan karyawan yang belum dilakukan penilaian berdasarkan periode tahun.	Untuk hasilnya dapat dilihat pada Gambar 4.4.
Pengguna memilih karyawan yang akan dilakukan penilaian kompetensi dalam melakukan pekerjaannya.	Sistem akan menampilkan informasi singkat tentang profil karyawan.	
Pengguna akan melakukan penilaian kompetensi karyawan dengan menekan tombol “ <i>Behavior Indicators Assessment</i> ”. Aksi tersebut dilakukan setelah memilih jenis dan nama kompetensi yang akan dinilai.	Sistem akan menampilkan indikator-indikator perilaku pada setiap tingkat kompetensi yang akan dinilai pada <i>Form Behavior Indicators Assessment</i> .	Untuk hasilnya dapat dilihat pada Gambar 4.5.
Pengguna akan melakukan penilaian kompetensi karyawan dengan menentukan tingkat kompeten pada setiap indikator perilaku.	Sistem akan menampilkan indikator perilaku dan tingkat kompeten. Tingkat kompeten tersebut akan dipilih sesuai dengan penilaian pengguna.	Untuk hasilnya dapat dilihat pada Gambar 4.6.
Pengguna akan menghentikan proses penilaian kompetensi karyawan pada tingkat kompetensi sesuai dengan aturan yang ditentukan.	Sistem akan menampilkan informasi mengenai tingkat kompetensi karyawan yang didapatkan sesuai dengan perhitungan yang telah ditentukan sebelumnya.	

<p>Pengguna akan menyimpan hasil penilaian kompetensi karyawan sesuai petunjuk sistem.</p>	<p>Sistem akan menampilkan tingkat kompetensi karyawan dan menampilkan hasil perhitungan kesenjangan antara tingkat kompetensi karyawan dengan tingkat kompetensi jabatan.</p>	
<p>Pengguna akan menyimpan data kompetensi karyawan beserta kebutuhan pelatihan yang akan dilaksanakan ke dalam database sesuai dengan petunjuk sistem.</p>	<p>Sistem berhasil menentukan kebutuhan pelatihan karyawan dan menyimpan data kompetensi karyawan beserta kebutuhan pelatihannya.</p>	
<p>Pengguna melihat laporan kebutuhan pelatihan dengan memilih sub menu "Department Analysis" pada menu "Report".</p>	<p>Sistem akan menampilkan form "Department Analysis".</p>	<p>Untuk hasilnya dapat dilihat pada Gambar 4.9.</p>
<p>Pengguna memilih kategori laporan yang ditampilkan berdasarkan nama departemen, nama kompetensi dan periode.</p>	<p>Sistem akan menampilkan laporan kebutuhan pelatihan bagi karyawan sesuai dengan kategori laporan yang dipilih.</p>	<p>Untuk hasilnya dapat dilihat pada Gambar 4.10.</p>
Alur Alternatif		
Aksi Pengguna	Respon Sistem	Hasil
<p>Pengguna melihat laporan kebutuhan pelatihan dengan memilih sub menu "Individual Analysis" pada menu "Report".</p>	<p>Sistem akan menampilkan form "Individual Analysis".</p>	<p>Untuk hasilnya dapat dilihat pada Gambar 4.11.</p>
<p>Pengguna memilih kategori laporan yang ditampilkan berdasarkan nama karyawan dan periode.</p>	<p>Sistem akan menampilkan laporan kebutuhan pelatihan bagi karyawan sesuai dengan nama karyawan dan periode yang dipilih.</p>	<p>Untuk hasilnya dapat dilihat pada Gambar 4.12.</p>

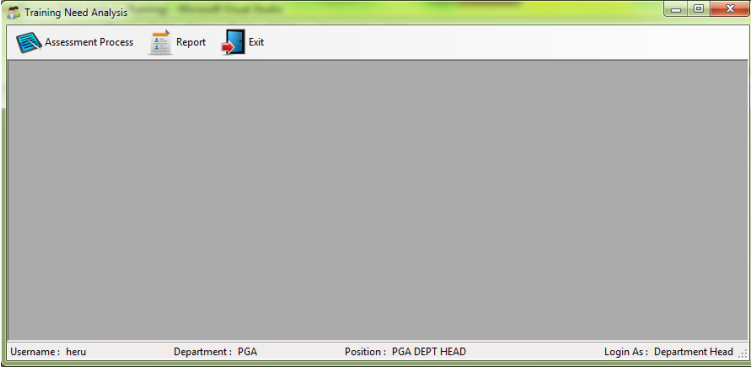
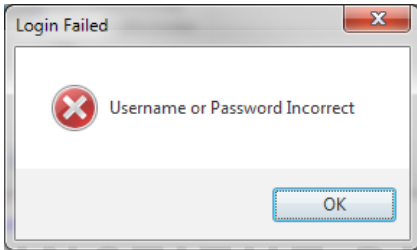
Alur Eksepsi		
Aksi Pengguna	Respon Sistem	Hasil
Pengguna tidak dapat melihat laporan kebutuhan pelatihan karyawan.	Sistem akan menampilkan pesan bahwa pengguna belum memilih kategori laporan yang akan di tampilkan.	
Kondisi Akhir		
Respon Sistem	Hasil	
Sistem berhasil menyimpan hasil penilaian kompetensi karyawan pada indikator perilaku.	Untuk hasilnya dapat dilihat pada Gambar 4.8.	
Sistem berhasil menyimpan data kompetensi karyawan beserta kebutuhan pelatihannya.	Untuk hasilnya dapat dilihat pada Gambar 4.10 dan Gambar 4.12.	

Setelah uji fungsional, langkah selanjutnya adalah uji non-fungsional. Uji non-fungsional ini disesuaikan dengan rancangan spesifikasi kebutuhan perangkat lunak yang sudah dibahas pada bab sebelumnya. Berikut adalah hasil uji non-fungsional pada fungsional merencanakan pengembangan karyawan:

1. Non-fungsional: *Security*

Tujuan dari uji non-fungsional ini adalah untuk menguji keamanan dari aplikasi yang telah dibuat. Aplikasi hanya dapat diakses oleh pengguna yang telah memiliki hak akses dengan memasukkan *username* dan *password*. *Username* dan *password* sebelumnya dapat diperoleh dari *administrator* dari aplikasi ini yang menjabat sebagai *IT officer*. Hasil uji dapat dilihat pada Tabel 4.21.

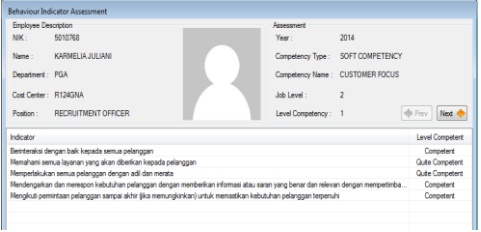
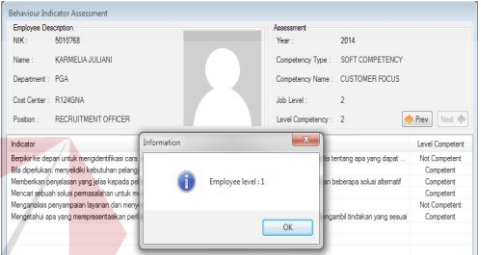
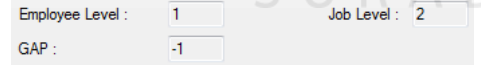
Tabel 4.21 Hasil Uji Non-Fungsional *Security* Pada Fungsional Merencanakan Pengembangan Karyawan

Keterangan	Hasil Sistem
<p>Otentikasi sistem terhadap <i>Department Head</i>.</p>	
<p>Pengguna sebagai <i>Department Head</i> salah memasukkan <i>username</i> atau <i>password</i> maupun keduanya.</p>	
<p>Kesimpulan</p>	
<p>Dari hasil uji non-fungsional <i>security</i>, dapat disimpulkan bahwa setelah pengguna melakukan <i>login</i>, sistem telah melakukan otentikasi yang sudah ditentukan. <i>Department Head</i> hanya dapat mengakses menu “<i>Assessment Process</i>” dan “<i>Report</i>”.</p>	

2. Non-fungsional: *Correctness*

Tujuan uji non-fungsional ini adalah untuk melakukan perhitungan secara akurat mengenai perhitungan dalam menentukan tingkat kompetensi karyawan dan tingkat kesenjangan dengan tingkat kompetensi jabatan. Hasil uji dapat dilihat pada Tabel 4.22.

Tabel 4.22 Hasil Uji Non-Fungsional *Correctness* Pada Fungsional Merencanakan Pengembangan Karyawan

Keterangan	Hasil Sistem	Hasil Manual															
<p>Karyawan tersebut dinyatakan telah berkompeten pada tingkat kompetensi ke-1 karena jumlah “Competent” pada indikator perilaku mencapai > 50%.</p>		<table border="1" data-bbox="1078 510 1366 607"> <thead> <tr> <th></th> <th>Jumlah</th> <th>Persentase</th> </tr> </thead> <tbody> <tr> <td>Competent</td> <td>3</td> <td>60%</td> </tr> <tr> <td>Quite Competent</td> <td>2</td> <td>40%</td> </tr> <tr> <td>Not Competent</td> <td>0</td> <td>0%</td> </tr> <tr> <td>Total</td> <td>5</td> <td>100%</td> </tr> </tbody> </table>		Jumlah	Persentase	Competent	3	60%	Quite Competent	2	40%	Not Competent	0	0%	Total	5	100%
	Jumlah	Persentase															
Competent	3	60%															
Quite Competent	2	40%															
Not Competent	0	0%															
Total	5	100%															
<p>Karyawan tersebut dinyatakan telah berkompeten pada tingkat kompetensi ke-1 karena tingkat kompetensi ke-2 memiliki jumlah “Not Competent” pada indikator perilaku sebesar > 0%.</p>		<table border="1" data-bbox="1078 952 1366 1048"> <thead> <tr> <th></th> <th>Jumlah</th> <th>Persentase</th> </tr> </thead> <tbody> <tr> <td>Competent</td> <td>4</td> <td>67%</td> </tr> <tr> <td>Quite Competent</td> <td>0</td> <td>0%</td> </tr> <tr> <td>Not Competent</td> <td>2</td> <td>33%</td> </tr> <tr> <td>Total</td> <td>6</td> <td>100%</td> </tr> </tbody> </table>		Jumlah	Persentase	Competent	4	67%	Quite Competent	0	0%	Not Competent	2	33%	Total	6	100%
	Jumlah	Persentase															
Competent	4	67%															
Quite Competent	0	0%															
Not Competent	2	33%															
Total	6	100%															
<p>Nilai <i>Gap</i> adalah -1, hasil tersebut didapat dari pengurangan antara nilai <i>Employee Level</i> dengan nilai <i>Job Level</i>.</p>		<table border="1" data-bbox="1078 1400 1366 1451"> <thead> <tr> <th>Employee Level</th> <th>Job Level</th> <th>Gap</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>2</td> <td>-1</td> </tr> </tbody> </table>	Employee Level	Job Level	Gap	1	2	-1									
Employee Level	Job Level	Gap															
1	2	-1															
Kesimpulan																	
<p>Dari hasil diatas, dapat disimpulkan bahwa hasil perhitungan sistem telah menghasilkan tingkat kompetensi karyawan dengan benar sesuai dengan hasil perhitungan secara manual. Selain itu, hasil perhitungan tingkat kesenjangan (<i>gap</i>) antara tingkat kompetensi karyawan dan tingkat kompetensi jabatan juga sesuai dengan konsep perhitungan <i>Gap Analysis</i>. Hasil perhitungan manual tersebut dilakukan pada Microsoft Excel.</p>																	

3. Non-fungsional: *Interface*

Tujuan uji non-fungsional ini adalah untuk melakukan pengujian pada desain antarmuka yang dapat membuat nyaman para pengguna aplikasi. Desain antarmuka mengacu pada antarmuka *windows* dan didukung dengan *icon-icon* pada semua jenis perintah. Adapun hasil ujinya dapat dilihat pada Tabel 4.23.

Tabel 4.23 Hasil Uji Non-Fungsional *Interface* Pada Fungsional Merencanakan Pengembangan Karyawan

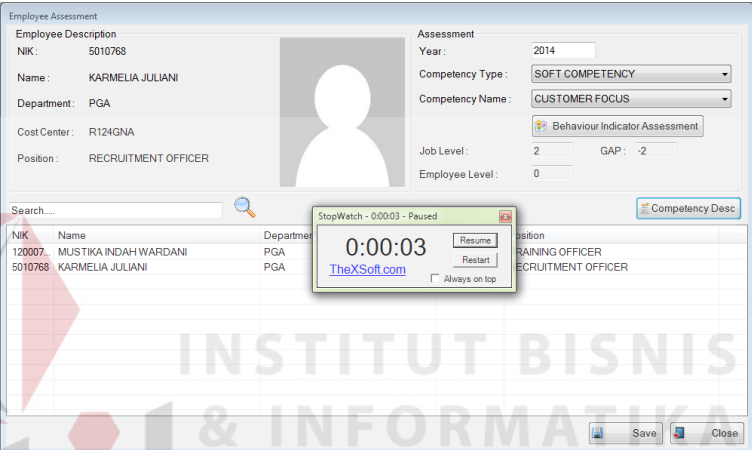
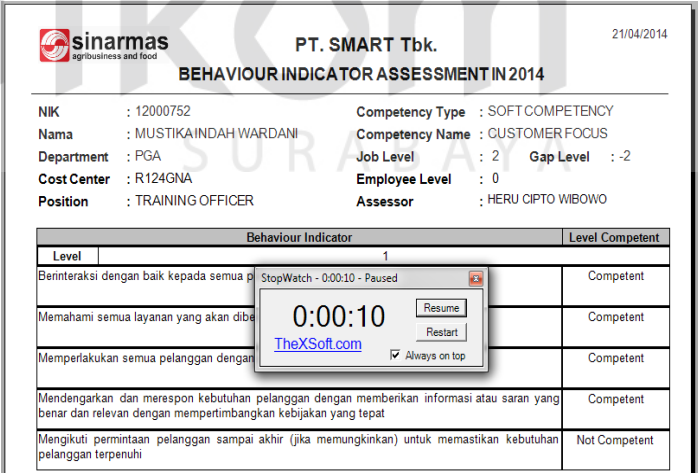
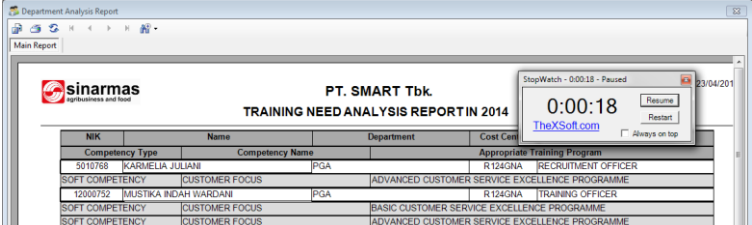
Keterangan	Hasil Sistem												
Warna yang digunakan pada <i>form</i> tidak terlalu mencolok.	 <table border="1" data-bbox="628 981 1353 1086"> <thead> <tr> <th>Indicator</th> <th>Level Competent</th> </tr> </thead> <tbody> <tr> <td>Berinteraksi dengan baik kepada semua pelanggan</td> <td>Competent</td> </tr> <tr> <td>Memahami semua layanan yang akan diberikan kepada pelanggan</td> <td>Quite Competent</td> </tr> <tr> <td>Memperlakukan semua pelanggan dengan adil dan merata</td> <td>Competent</td> </tr> <tr> <td>Mendengarkan dan merespon kebutuhan pelanggan dengan memberikan informasi atau saran yang benar dan relevan dengan mempertim...</td> <td>Competent</td> </tr> <tr> <td>Mengikuti permintaan pelanggan sampai akhir (jika memungkinkan) untuk memastikan kebutuhan pelanggan terpenuhi</td> <td>Competent</td> </tr> </tbody> </table>	Indicator	Level Competent	Berinteraksi dengan baik kepada semua pelanggan	Competent	Memahami semua layanan yang akan diberikan kepada pelanggan	Quite Competent	Memperlakukan semua pelanggan dengan adil dan merata	Competent	Mendengarkan dan merespon kebutuhan pelanggan dengan memberikan informasi atau saran yang benar dan relevan dengan mempertim...	Competent	Mengikuti permintaan pelanggan sampai akhir (jika memungkinkan) untuk memastikan kebutuhan pelanggan terpenuhi	Competent
Indicator	Level Competent												
Berinteraksi dengan baik kepada semua pelanggan	Competent												
Memahami semua layanan yang akan diberikan kepada pelanggan	Quite Competent												
Memperlakukan semua pelanggan dengan adil dan merata	Competent												
Mendengarkan dan merespon kebutuhan pelanggan dengan memberikan informasi atau saran yang benar dan relevan dengan mempertim...	Competent												
Mengikuti permintaan pelanggan sampai akhir (jika memungkinkan) untuk memastikan kebutuhan pelanggan terpenuhi	Competent												
<i>Icon-icon</i> yang digunakan pada jenis perintah menarik dan mudah dalam mendefinisikan perintah.													
Kesimpulan													
Sistem dirancang dengan desain yang disesuaikan dengan kebutuhan pengguna, sehingga dapat memberikan kemudahan dan kenyamanan dalam menjalankan aplikasi ini.													

4. Non-fungsional: *Performance*

Tujuan uji non-fungsional ini adalah untuk menguji kinerja dari aplikasi yang telah dibuat. Untuk melakukan penentuan tingkat kompetensi karyawan beserta perhitungan tingkat kesenjangan, aplikasi harus mampu melaksanakannya

dalam waktu tidak lebih dari 5 detik. Selain itu, dalam proses untuk mendapatkan laporan hasil penilaian kompetensi, aplikasi harus mampu menampilkan laporan tersebut dengan waktu tidak lebih dari 30 detik. Adapun hasil ujinya dapat dilihat pada Tabel 4.24.

Tabel 4.24 Hasil Uji Non-Fungsional *Performance* Pada Fungsional Merencanakan Pengembangan Karyawan

Keterangan	Hasil Sistem
<p>Waktu yang dibutuhkan dalam melakukan penentuan tingkat kompetensi karyawan beserta perhitungan tingkat kesenjangan adalah 3 detik.</p>	
<p>Waktu yang dibutuhkan dalam menampilkan laporan hasil penilaian mencapai 10 detik.</p>	
<p>Waktu yang dibutuhkan dalam menampilkan laporan kebutuhan pelatihan karyawan adalah 18 detik.</p>	

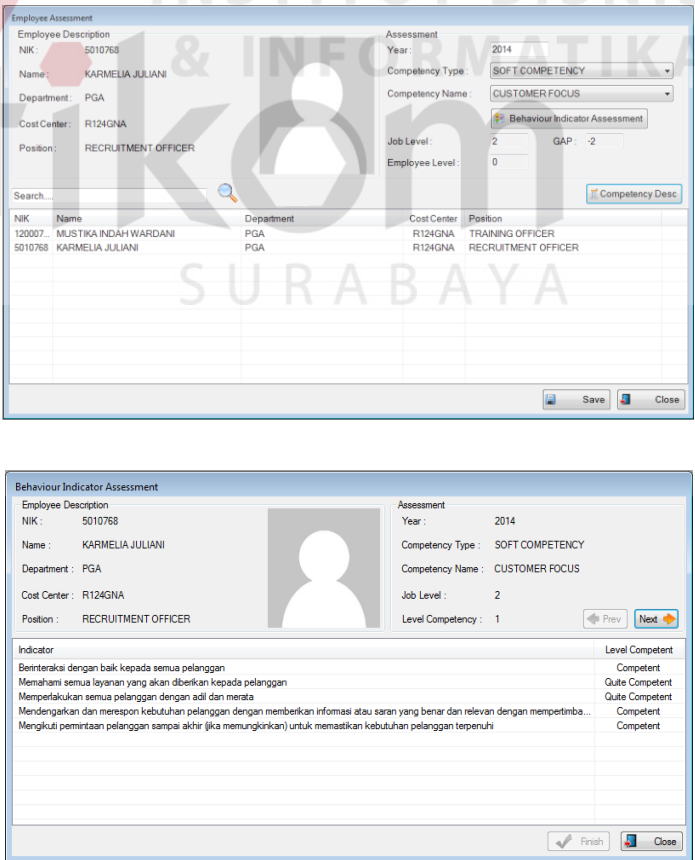
Kesimpulan

Uji non-fungsional ini dilakukan dengan menggunakan komputer yang memiliki spesifikasi lebih rendah dari kebutuhan minimum perangkat lunak yang disarankan. Hasil pengujian ini menyimpulkan bahwa sistem mampu melakukan proses perhitungan lebih cepat dari yang diharapkan dan mampu menampilkan laporan secara cepat dari yang diharapkan.

5. Non-fungsional: *Operability*

Tujuan uji non-fungsional ini adalah untuk menguji kemudahan dalam mengoperasikan aplikasi yang telah dibuat. Uji akan yang dilakukan terhadap posisi atau letak tombol perintah, kolom-kolom dan teks yang terdapat pada *form*. Uji dapat dilihat pada Tabel 4.25.

Tabel 4.25 Hasil Uji Non-Fungsional *Operability* Pada Fungsional Merencanakan Pengembangan Karyawan

Keterangan	Hasil Sistem
<p>Tata letak keterangan maupun tombol-tombol diposisikan rapi dan dikelompokkan berdasarkan fungsinya.</p>	

Bahasa digunakan adalah bahasa yang disesuaikan dengan kebutuhan perusahaan.

NIK	Name	Department	Cost Center	Position
120007	MUSTIKA INDAH WARDANI	PGA	R124GNA	TRAINING OFFICER
5010768	KARMEIA JULIANI	PGA	R124GNA	RECRUITMENT OFFICER

Kesimpulan

Uji non-fungsional ini menyimpulkan bahwa posisi keterangan maupun perintah-perintah yang telah tersedia dikelompokkan sesuai fungsinya dan bahasa yang digunakan telah disesuaikan dengan kebutuhan perusahaan. Hal tersebut sudah dapat memudahkan pengguna dalam mengoperasikan aplikasi yang telah dibuat.

4.4.2 Uji Coba Pada *Training Officer*

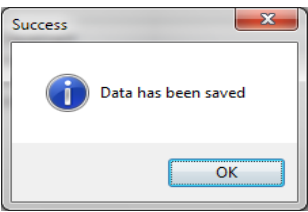
Uji fungsional yang dilakukan *training officer* adalah sebagai berikut:

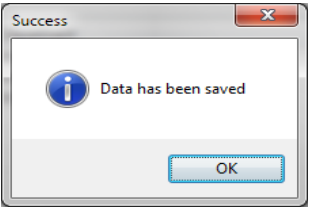
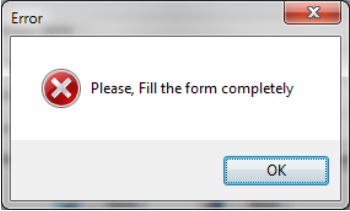
1. Melakukan *Update Active File*

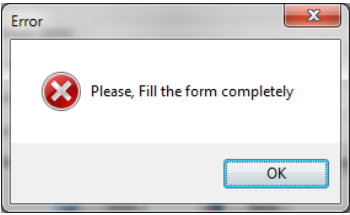
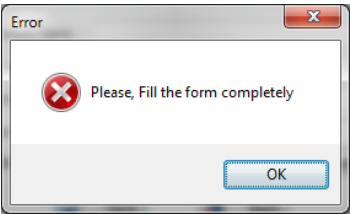
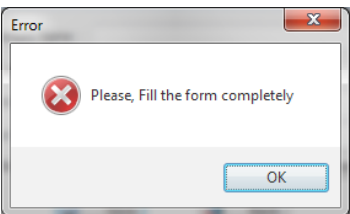
Mengenai uji fungsional melakukan *update active file* lebih detail dapat dilihat pada Tabel 4.26.

Tabel 4.26 Hasil Uji Fungsional Melakukan *Update Active File*

Fungsional	Melakukan <i>Update Active File</i>		
Stakeholder	<i>Training officer</i>		
Alur Normal			
Aksi Pengguna	Respon Sistem		Hasil
Pengguna memilih sub menu “ <i>Department</i> ” pada menu “ <i>Maintenance File</i> ”.	Sistem akan menampilkan tabel yang berisi data jabatan.	<i>form</i> yang	Untuk hasilnya dapat dilihat pada Gambar 4.13.
Pengguna akan memasukkan data jabatan dengan menekan tombol “ <i>Add</i> ”.	Sistem akan menampilkan “ <i>Add Position</i> ”.	<i>form</i>	Untuk hasilnya dapat dilihat pada Gambar 4.14.

<p>Pengguna akan menyimpan data jabatan dengan menekan tombol “Save”.</p>	<p>Sistem akan menampilkan pemberitahuan bahwa data jabatan telah tersimpan ke dalam <i>database</i>.</p>	
<p>Pengguna memilih sub menu “Employee” pada menu “Maintenance File”.</p>	<p>Sistem akan menampilkan <i>form</i> dengan tabel yang berisi data karyawan.</p>	<p>Untuk hasilnya dapat dilihat pada Gambar 4.15.</p>
<p>Pengguna akan memasukkan data karyawan dengan menekan tombol “Add”.</p>	<p>Sistem akan menampilkan <i>form</i> “Add Employee”.</p>	<p>Untuk hasilnya dapat dilihat pada Gambar 4.16.</p>
<p>Pengguna akan menyimpan data karyawan dengan menekan tombol “Save”.</p>	<p>Sistem akan menampilkan pemberitahuan bahwa data karyawan telah tersimpan ke dalam <i>database</i>.</p>	
<p>Pengguna memilih sub menu “Competency” pada menu “Maintenance File”.</p>	<p>Sistem akan menampilkan <i>form</i> dengan tabel-tabel yang berisi data kompetensi dan data indikator perilaku.</p>	<p>Untuk hasilnya dapat dilihat pada Gambar 4.17.</p>
<p>Pengguna akan memasukkan data kompetensi dengan menekan tombol “Add”.</p>	<p>Sistem akan menampilkan <i>form</i> “Add Competency”.</p>	<p>Untuk hasilnya dapat dilihat pada Gambar 4.18.</p>
<p>Pengguna menekan tombol “Save” untuk menyimpan data kompetensi pada <i>form</i> “Add Competency”.</p>	<p>Sistem akan menampilkan pemberitahuan bahwa data kompetensi telah tersimpan ke dalam <i>database</i>.</p>	
<p>Pengguna akan memasukkan data indikator perilaku dengan menekan tombol “Add” setelah memilih nama kompetensi dan tingkat kompetensi.</p>	<p>Sistem akan menampilkan <i>form</i> “Add Behavior Indicator”.</p>	<p>Untuk hasilnya dapat dilihat pada Gambar 4.18.</p>

<p>Pengguna menekan tombol “<i>Save</i>” untuk menyimpan data indikator perilaku setiap kompetensi pada <i>form “Add Behavior Indicator”</i>.</p>	<p>Sistem akan menampilkan pemberitahuan bahwa data indikator perilaku setiap kompetensi telah tersimpan ke dalam <i>database</i>.</p>	
Alur Alternatif		
Aksi Pengguna	Respon Sistem	Hasil
<p>Pengguna mengubah data jabatan dengan menekan tombol “<i>Edit</i>” setelah pengguna memilih data jabatan yang akan dirubah</p>	<p>Sistem akan menampilkan <i>form “Edit Position”</i>.</p>	<p>Untuk hasilnya dapat dilihat pada Gambar 4.14.</p>
<p>Pengguna mengubah data karyawan dengan menekan tombol “<i>Edit</i>” setelah pengguna memilih data karyawan yang akan dirubah.</p>	<p>Sistem akan menampilkan <i>form “Edit Employee”</i>.</p>	<p>Untuk hasilnya dapat dilihat pada Gambar 4.16.</p>
<p>Pengguna mengubah data kompetensi dengan menekan tombol “<i>Edit</i>” setelah pengguna memilih data kompetensi yang akan dirubah.</p>	<p>Sistem akan menampilkan <i>form “Edit Competency”</i>.</p>	<p>Untuk hasilnya dapat dilihat pada Gambar 4.18.</p>
<p>Pengguna mengubah data indikator perilaku dengan menekan tombol “<i>Edit</i>” setelah pengguna memilih data indikator perilaku pada setiap tingkat kompetensi yang dipilih.</p>	<p>Sistem akan menampilkan <i>form “Edit Behavior Indicator”</i>.</p>	<p>Untuk hasilnya dapat dilihat pada Gambar 4.19.</p>
Alur Eksepsi		
Aksi Pengguna	Respon Sistem	Hasil
<p>Pengguna tidak dapat menyimpan data jabatan ke dalam <i>database</i>.</p>	<p>Sistem akan menampilkan pesan bahwa <i>form</i> belum terisi sesuai dengan syarat pengisian data.</p>	

Pengguna tidak dapat menyimpan data karyawan ke dalam <i>database</i> .	Sistem akan menampilkan pesan bahwa <i>form</i> belum terisi sesuai dengan syarat pengisian data.	
Pengguna tidak dapat menyimpan data kompetensi ke dalam <i>database</i> .	Sistem akan menampilkan pesan bahwa <i>form</i> belum terisi sesuai dengan syarat pengisian data.	
Pengguna tidak dapat menyimpan data indikator perilaku setiap tingkat kompetensi ke dalam <i>database</i> .	Sistem akan menampilkan pesan bahwa <i>form</i> belum terisi sesuai dengan syarat pengisian data.	
Kondisi Akhir		
Respon Sistem	Hasil	
Sistem berhasil menyimpan data jabatan ke dalam <i>database</i> .	Untuk hasilnya dapat dilihat pada Gambar 4.13.	
Sistem berhasil menyimpan data karyawan ke dalam <i>database</i> .	Untuk hasilnya dapat dilihat pada Gambar 4.15.	
Sistem berhasil menyimpan data kompetensi ke dalam <i>database</i> .	Untuk hasilnya dapat dilihat pada Gambar 4.17.	
Sistem berhasil menyimpan data indikator perilaku setiap tingkat kompetensi ke dalam <i>database</i> .	Untuk hasilnya dapat dilihat pada Gambar 4.17.	

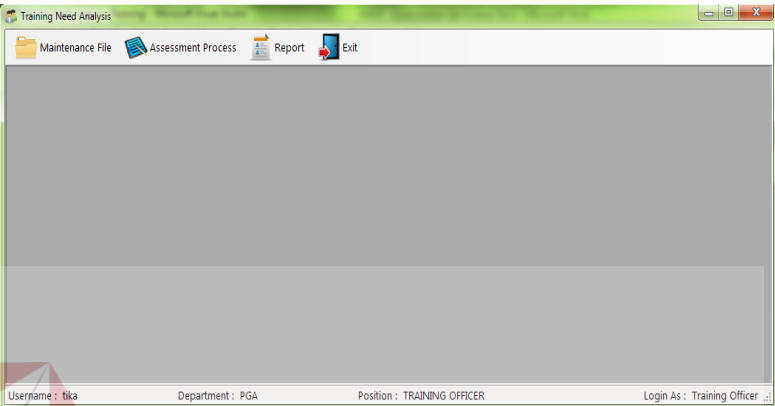
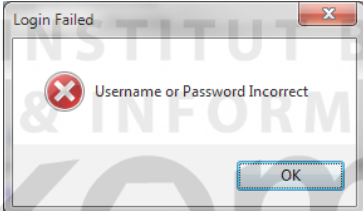
Setelah uji fungsional, langkah selanjutnya adalah uji non-fungsional. Berikut adalah hasil uji non-fungsional pada fungsional melakukan *update active file*:

a. Non-fungsional: *Security*

Tujuan dari uji non-fungsional ini adalah untuk menguji keamanan dari aplikasi yang telah dibuat. Aplikasi hanya dapat diakses oleh pengguna yang telah memiliki hak akses dengan memasukkan *username* dan *password*. *Username* dan *password* sebelumnya dapat diperoleh dari *administrator*

dari aplikasi ini yang menjabat sebagai *IT officer*. Hasil uji dapat dilihat pada Tabel 4.27.

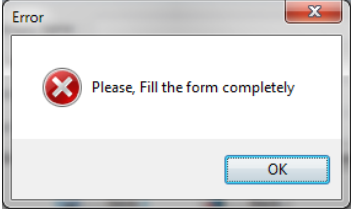
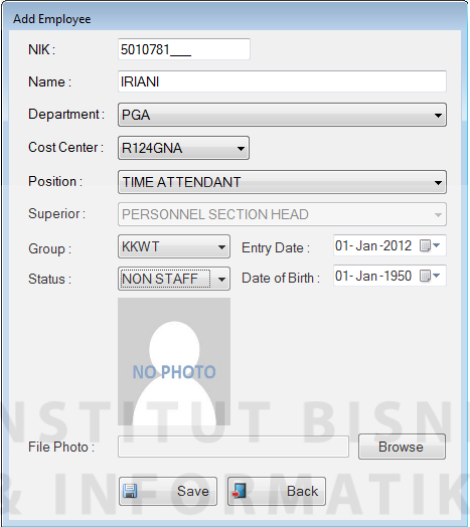
Tabel 4.27 Hasil Uji Non-Fungsional *Security* Pada Fungsional Melakukan *Update Active File*

Keterangan	Hasil Sistem
Otentikasi sistem terhadap <i>Training Officer</i> .	
Pengguna sebagai <i>Training Officer</i> salah memasukkan <i>username</i> atau <i>password</i> maupun keduanya.	
Kesimpulan	
Dari hasil uji non-fungsional <i>security</i> , dapat disimpulkan bahwa setelah pengguna melakukan <i>login</i> , sistem telah melakukan otentikasi yang sudah ditentukan. <i>Training Officer</i> dapat mengakses menu " <i>Maintenance File</i> ", " <i>Assessment Process</i> " dan " <i>Report</i> ".	

b. Non-fungsional: *Correctness*

Tujuan dari non-fungsional ini adalah memvalidasi data-data pada saat akan melakukan penyimpanan data. Hasil uji dapat dilihat pada Tabel 4.28.

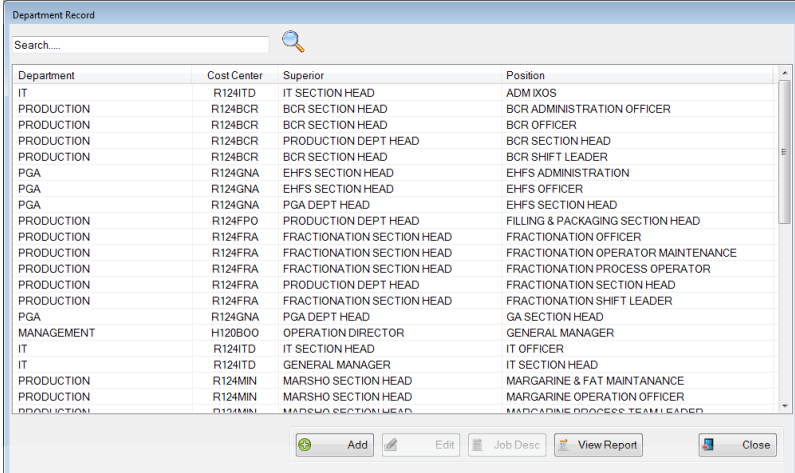
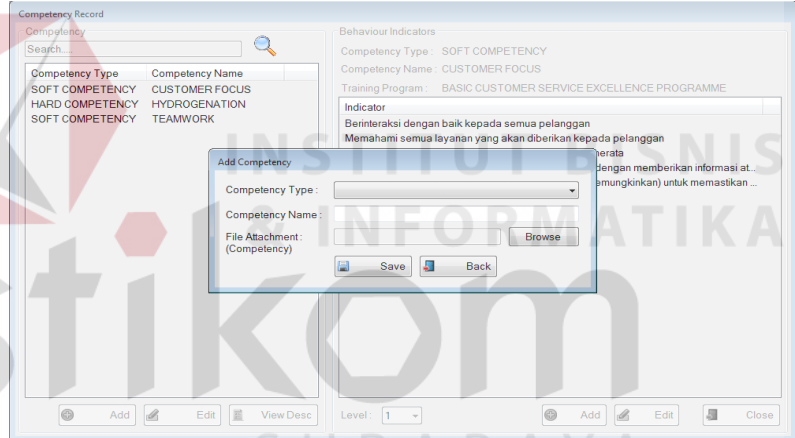
Tabel 4.28 Hasil Uji Non-Fungsional *Correctness* Pada Fungsional Melakukan *Update Active File*

Keterangan	Hasil Sistem
<p>Sistem berhasil menampilkan pesan yang mengatakan bahwa <i>form</i> harus diisi secara lengkap dan <i>field-field</i> tidak boleh ada yang kosong.</p>	
<p>Kolom-kolom isian telah disesuaikan dengan kebutuhan tipe dan jenis data yang akan diisikan.</p>	
Kesimpulan	
<p>Dari hasil uji coba dalam mengisi <i>form</i> diatas telah diketahui bahwa <i>field-field</i> yang tersedia pada <i>form</i> tersebut pengisiannya telah disesuaikan dengan tipe dan jenis data yang akan diisikan. Sehingga data tidak akan dapat masuk jika tidak sesuai. Selain itu, sistem tidak akan melakukan penyimpanan data ke dalam <i>database</i> jika <i>field-field</i> tersebut tidak terisi.</p>	

c. Non-fungsional: *Interface*

Tujuan uji non-fungsional ini adalah untuk melakukan pengujian pada desain antarmuka yang dapat membuat nyaman para pengguna aplikasi. Desain antarmuka mengacu pada antarmuka *windows* dan didukung dengan *icon-icon* pada semua jenis perintah. Adapun hasil ujinya dapat dilihat pada Tabel 4.29.

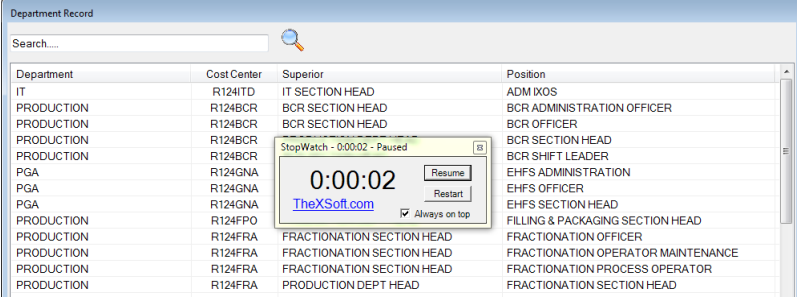
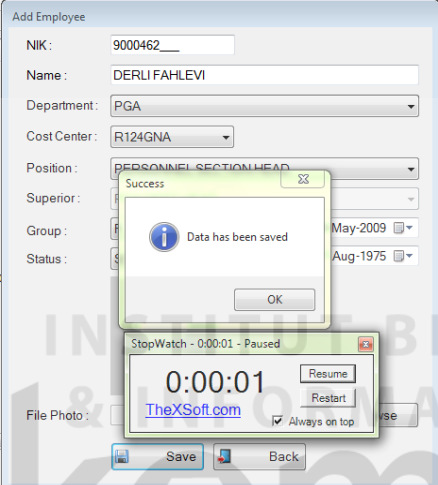
Tabel 4.29 Hasil Uji Non-Fungsional *Interface* Pada Fungsional Melakukan *Update Active File*

Keterangan	Hasil Sistem
<p>Icon-icon yang digunakan pada jenis perintah menarik dan mudah dalam mendefinisikan perintah.</p>	
<p>Warna yang digunakan pada form tidak terlalu mencolok.</p>	
Kesimpulan	
<p>Sistem dirancang dengan desain yang disesuaikan dengan kebutuhan pengguna, sehingga dapat memberikan kemudahan dan kenyamanan dalam menjalankan aplikasi ini.</p>	

d. Non-fungsional: *Performance*

Tujuan uji non-fungsional ini adalah untuk menguji kinerja dari aplikasi yang telah dibuat. Untuk melakukan penyimpanan data dan menampilkan data, aplikasi harus mampu melaksanakannya dengan cepat dalam waktu tidak lebih dari 5 detik. Adapun hasil ujinya dapat dilihat pada Tabel 4.30.

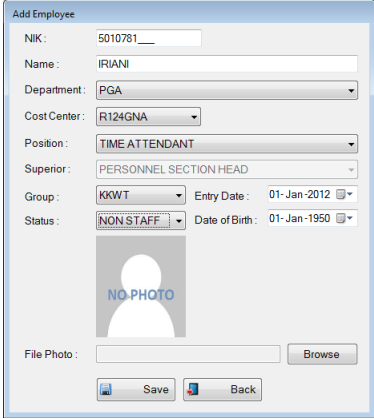
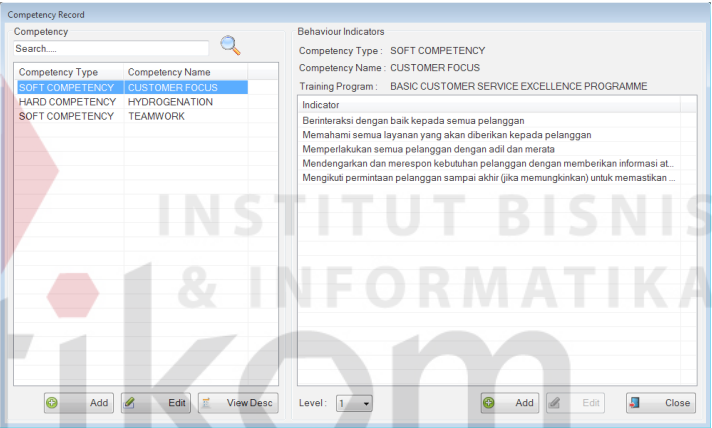
Tabel 4.30 Hasil Uji Non-Fungsional *Performance* Pada Fungsional Melakukan *Update Active File*

Keterangan	Hasil Sistem																																																								
<p>Waktu yang dibutuhkan dalam menampilkan data yang telah tersedia di dalam <i>database</i> adalah 2 detik.</p>	 <table border="1" data-bbox="564 398 1364 696"> <thead> <tr> <th>Department</th> <th>Cost Center</th> <th>Superior</th> <th>Position</th> </tr> </thead> <tbody> <tr><td>IT</td><td>R124ITD</td><td>IT SECTION HEAD</td><td>ADM IXOS</td></tr> <tr><td>PRODUCTION</td><td>R124BCR</td><td>BCR SECTION HEAD</td><td>BCR ADMINISTRATION OFFICER</td></tr> <tr><td>PRODUCTION</td><td>R124BCR</td><td>BCR SECTION HEAD</td><td>BCR OFFICER</td></tr> <tr><td>PRODUCTION</td><td>R124BCR</td><td>BCR SECTION HEAD</td><td>BCR SECTION HEAD</td></tr> <tr><td>PRODUCTION</td><td>R124BCR</td><td>BCR SECTION HEAD</td><td>BCR SHIFT LEADER</td></tr> <tr><td>PGA</td><td>R124GNA</td><td>FRACTIONATION SECTION HEAD</td><td>EHFS ADMINISTRATION</td></tr> <tr><td>PGA</td><td>R124GNA</td><td>FRACTIONATION SECTION HEAD</td><td>EHFS OFFICER</td></tr> <tr><td>PGA</td><td>R124GNA</td><td>FRACTIONATION SECTION HEAD</td><td>EHFS SECTION HEAD</td></tr> <tr><td>PRODUCTION</td><td>R124FPO</td><td>FRACTIONATION SECTION HEAD</td><td>FILLING & PACKAGING SECTION HEAD</td></tr> <tr><td>PRODUCTION</td><td>R124FRA</td><td>FRACTIONATION SECTION HEAD</td><td>FRACTIONATION OFFICER</td></tr> <tr><td>PRODUCTION</td><td>R124FRA</td><td>FRACTIONATION SECTION HEAD</td><td>FRACTIONATION OPERATOR MAINTENANCE</td></tr> <tr><td>PRODUCTION</td><td>R124FRA</td><td>FRACTIONATION SECTION HEAD</td><td>FRACTIONATION PROCESS OPERATOR</td></tr> <tr><td>PRODUCTION</td><td>R124FRA</td><td>PRODUCTION DEPT HEAD</td><td>FRACTIONATION SECTION HEAD</td></tr> </tbody> </table>	Department	Cost Center	Superior	Position	IT	R124ITD	IT SECTION HEAD	ADM IXOS	PRODUCTION	R124BCR	BCR SECTION HEAD	BCR ADMINISTRATION OFFICER	PRODUCTION	R124BCR	BCR SECTION HEAD	BCR OFFICER	PRODUCTION	R124BCR	BCR SECTION HEAD	BCR SECTION HEAD	PRODUCTION	R124BCR	BCR SECTION HEAD	BCR SHIFT LEADER	PGA	R124GNA	FRACTIONATION SECTION HEAD	EHFS ADMINISTRATION	PGA	R124GNA	FRACTIONATION SECTION HEAD	EHFS OFFICER	PGA	R124GNA	FRACTIONATION SECTION HEAD	EHFS SECTION HEAD	PRODUCTION	R124FPO	FRACTIONATION SECTION HEAD	FILLING & PACKAGING SECTION HEAD	PRODUCTION	R124FRA	FRACTIONATION SECTION HEAD	FRACTIONATION OFFICER	PRODUCTION	R124FRA	FRACTIONATION SECTION HEAD	FRACTIONATION OPERATOR MAINTENANCE	PRODUCTION	R124FRA	FRACTIONATION SECTION HEAD	FRACTIONATION PROCESS OPERATOR	PRODUCTION	R124FRA	PRODUCTION DEPT HEAD	FRACTIONATION SECTION HEAD
Department	Cost Center	Superior	Position																																																						
IT	R124ITD	IT SECTION HEAD	ADM IXOS																																																						
PRODUCTION	R124BCR	BCR SECTION HEAD	BCR ADMINISTRATION OFFICER																																																						
PRODUCTION	R124BCR	BCR SECTION HEAD	BCR OFFICER																																																						
PRODUCTION	R124BCR	BCR SECTION HEAD	BCR SECTION HEAD																																																						
PRODUCTION	R124BCR	BCR SECTION HEAD	BCR SHIFT LEADER																																																						
PGA	R124GNA	FRACTIONATION SECTION HEAD	EHFS ADMINISTRATION																																																						
PGA	R124GNA	FRACTIONATION SECTION HEAD	EHFS OFFICER																																																						
PGA	R124GNA	FRACTIONATION SECTION HEAD	EHFS SECTION HEAD																																																						
PRODUCTION	R124FPO	FRACTIONATION SECTION HEAD	FILLING & PACKAGING SECTION HEAD																																																						
PRODUCTION	R124FRA	FRACTIONATION SECTION HEAD	FRACTIONATION OFFICER																																																						
PRODUCTION	R124FRA	FRACTIONATION SECTION HEAD	FRACTIONATION OPERATOR MAINTENANCE																																																						
PRODUCTION	R124FRA	FRACTIONATION SECTION HEAD	FRACTIONATION PROCESS OPERATOR																																																						
PRODUCTION	R124FRA	PRODUCTION DEPT HEAD	FRACTIONATION SECTION HEAD																																																						
<p>Waktu yang dibutuhkan dalam menyimpan data ke dalam <i>database</i> adalah 1 detik.</p>																																																									
Kesimpulan																																																									
<p>Uji non-fungsional ini dilakukan dengan menggunakan komputer yang memiliki spesifikasi lebih rendah dari kebutuhan minimum perangkat lunak yang disarankan. Hasil pengujian ini menyimpulkan bahwa sistem mampu menampilkan data dari <i>database</i> dan menyimpan ke dalam <i>database</i> secara cepat sesuai dengan apa yang diharapkan.</p>																																																									

e. Non-fungsional: *Operability*

Tujuan uji non-fungsional ini adalah untuk menguji kemudahan dalam mengoperasikan aplikasi yang telah dibuat. Uji akan yang dilakukan terhadap posisi atau letak tombol perintah, kolom-kolom dan teks yang terdapat pada *form*. Uji dapat dilihat pada Tabel 4.31.

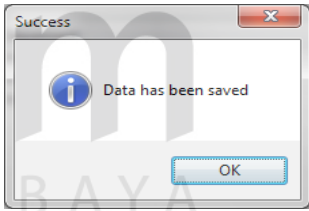
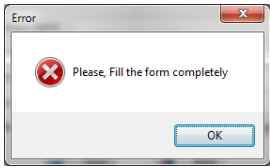
Tabel 4.31 Hasil Uji Non-Fungsional *Operability* Pada Fungsional Melakukan *Update Active File*

Keterangan	Hasil Sistem
<p>Bahasa digunakan adalah bahasa yang disesuaikan dengan kebutuhan perusahaan.</p>	
<p>Tata letak keterangan maupun tombol-tombol perintah diposisikan secara rapi dan dikelompokkan berdasarkan fungsinya.</p>	
<p>Kesimpulan</p>	
<p>Uji non-fungsional ini menyimpulkan bahwa posisi keterangan maupun perintah-perintah yang telah tersedia dikelompokkan sesuai fungsinya dan bahasa yang digunakan telah disesuaikan dengan kebutuhan perusahaan. Hal tersebut sudah dapat memudahkan pengguna dalam mengoperasikan aplikasi yang telah dibuat.</p>	

2. Membuat Standar Kompetensi Jabatan

Mengenai uji fungsional membuat standar kompetensi pada setiap jabatan lebih detail dapat dilihat pada Tabel 4.32.

Tabel 4.32 Hasil Uji Fungsional Membuat Standar Kompetensi Jabatan

Fungsional	Membuat Standar Kompetensi Jabatan	
Stakeholder	Training officer	
Alur Normal		
Aksi Pengguna	Respon Sistem	Hasil
Pengguna memilih sub menu “Standard Competency” pada menu “Maintenance File”.	Sistem akan menampilkan form dengan tabel yang berisi data standar kompetensi.	Untuk hasilnya dapat dilihat pada Gambar 4.20.
Pengguna memilih nama departemen, cost center dan jabatan untuk menambahkan kebutuhan standar kompetensinya.	Sistem akan menampilkan data standar kompetensi pada jabatan yang telah dipilih oleh pengguna.	Untuk hasilnya dapat dilihat pada Gambar 4.20.
Pengguna menambah standar kompetensi beserta tingkat kompetensi yang dibutuhkan oleh jabatan dengan menekan tombol “Add”.	Sistem akan menampilkan form “Add Standard Competency”.	Untuk hasilnya dapat dilihat pada Gambar 4.21.
Pengguna menyimpan data standar kompetensi jabatan dengan menekan tombol “Save”.	Sistem akan menyimpan data standar kompetensi yang dibutuhkan oleh jabatan.	
Alur Alternatif		
Aksi Pengguna	Respon Sistem	Hasil
-	-	-
Alur Eksepsi		
Aksi Pengguna	Respon Sistem	Hasil
Pengguna tidak dapat menyimpan data standar kompetensi yang akan ditambahkan ke jabatan.	Sistem akan menampilkan pesan yang menunjukkan bahwa form belum terisi sepenuhnya.	
Kondisi Akhir		
Respon Sistem	Hasil	
Sistem berhasil menyimpan data standar kompetensi pada setiap jabatan ke dalam database.	Untuk hasilnya dapat dilihat pada Gambar 4.20.	

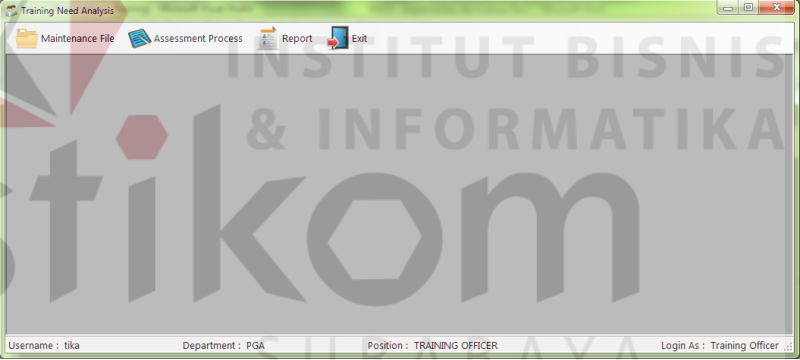
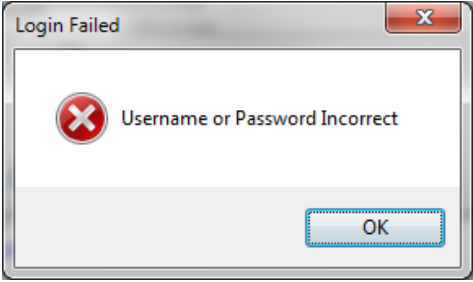
Setelah uji fungsional, langkah selanjutnya adalah uji non-fungsional. Berikut adalah hasil uji non-fungsional pada fungsional membuat standar kompetensi jabatan:

a. Non-fungsional: *Security*

Tujuan dari uji non-fungsional ini adalah untuk menguji keamanan dari aplikasi yang telah dibuat. Aplikasi hanya dapat diakses oleh pengguna yang telah memiliki hak akses dengan memasukkan *username* dan *password*.

Hasil uji dapat dilihat pada Tabel 4.33.

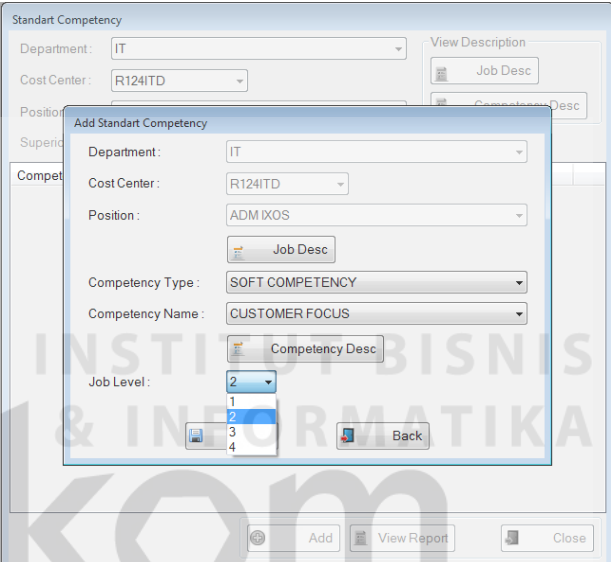
Tabel 4.33 Hasil Uji Non-Fungsional *Security* Pada Fungsional Membuat Standar Kompetensi Jabatan

Keterangan	Hasil Sistem
<p>Otentikasi sistem terhadap <i>Training Officer</i>.</p>	
<p>Pengguna sebagai <i>Training Officer</i> salah memasukkan <i>username</i> atau <i>password</i> maupun keduanya.</p>	
Kesimpulan	
<p>Dari hasil uji non-fungsional <i>security</i>, dapat disimpulkan bahwa setelah pengguna melakukan <i>login</i>, sistem telah melakukan otentikasi yang sudah ditentukan. <i>Training Officer</i> dapat mengakses menu “<i>Maintenance File</i>”, “<i>Assessment Process</i>” dan “<i>Report</i>”.</p>	

b. Non-fungsional: *Correctness*

Tujuan dari non-fungsional ini adalah memvalidasi data-data pada saat akan melakukan penyimpanan data supaya tidak ada kesalahan *input* akibat *human error*. Hasil uji dapat dilihat pada Tabel 4.34.

Tabel 4.34 Hasil Uji Non-Fungsional *Correctness* Pada Fungsional Membuat Standar Kompetensi Jabatan

Keterangan	Hasil Sistem
<p>Kolom-kolom isian telah disesuaikan dengan kebutuhan tipe dan jenis data yang akan diisikan.</p>	
<p>Sistem berhasil menampilkan pesan yang mengatakan bahwa <i>form</i> harus diisi secara lengkap dan <i>field-filed</i> tidak boleh ada yang kosong.</p>	
<p>Kesimpulan</p>	
<p>Dari hasil uji coba dalam mengisi <i>form</i> diatas telah diketahui bahwa <i>field-field</i> yang tersedia pada <i>form</i> tersebut pengisiannya telah disesuaikan dengan tipe dan jenis data yang akan diisikan. Sehingga data tidak akan dapat masuk jika tidak sesuai. Selain itu, sistem tidak akan melakukan penyimpanan data ke dalam <i>database</i> jika <i>field-field</i> tersebut tidak terisi.</p>	

c. Non-fungsional: *Interface*

Tujuan uji non-fungsional ini adalah untuk melakukan pengujian pada desain antarmuka yang dapat membuat nyaman para pengguna. Desain antarmuka mengacu pada antarmuka *windows* dan didukung dengan *icon-icon* pada semua jenis perintah. Adapun hasil ujinya dapat dilihat pada Tabel 4.35.

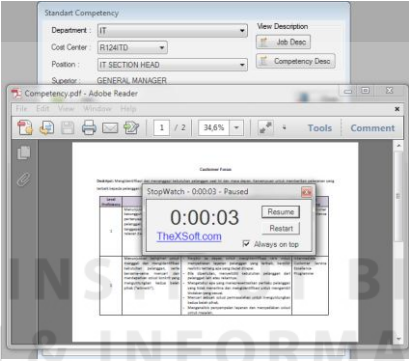
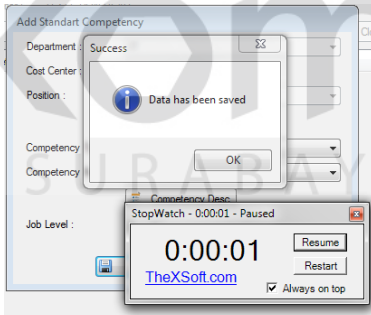
Tabel 4.35 Hasil Uji Non-Fungsional *Interface* Pada Fungsional Membuat Standar Kompetensi

Keterangan	Hasil Sistem
Warna yang digunakan pada <i>form</i> tidak terlalu mencolok.	
Icon-icon yang digunakan pada jenis perintah menarik dan mudah dalam mendefinisikan perintah.	
Kesimpulan	
Sistem dirancang dengan desain yang disesuaikan dengan kebutuhan pengguna, sehingga dapat memberikan kemudahan dan kenyamanan dalam menjalankan aplikasi ini.	

d. Non-fungsional: *Performance*

Tujuan uji non-fungsional ini adalah untuk menguji kinerja dari aplikasi yang telah dibuat. Untuk melakukan penyimpanan data dan menampilkan data, aplikasi harus mampu melaksanakannya dengan cepat dalam waktu tidak lebih dari 5 detik. Adapun hasil ujinya dapat dilihat pada Tabel 4.36.

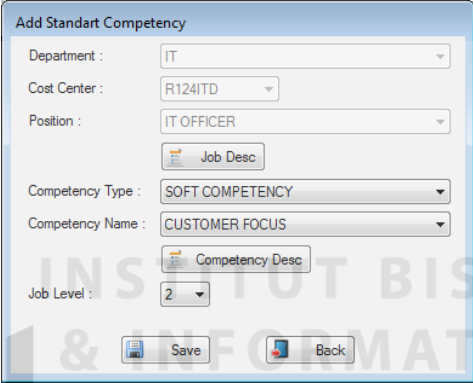
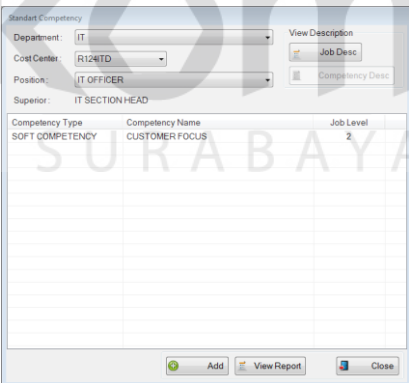
Tabel 4.36 Hasil Uji Non-Fungsional *Performance* Pada Fungsional Membuat Standar Kompetensi Jabatan

Keterangan	Hasil Sistem
<p>Waktu yang dibutuhkan dalam menampilkan dokumen kompetensi yang tersedia dari <i>database</i> adalah 3 detik.</p>	
<p>Waktu yang dibutuhkan dalam menyimpan data standar kompetensi yang dibutuhkan oleh jabatan ke dalam <i>database</i> adalah 1 detik.</p>	
Kesimpulan	
<p>Uji non-fungsional ini dilakukan dengan menggunakan komputer yang memiliki spesifikasi lebih rendah dari kebutuhan minimum perangkat lunak yang disarankan. Hasil pengujian ini menyimpulkan bahwa sistem mampu menampilkan data dari <i>database</i> dan menyimpan ke dalam <i>database</i> secara cepat sesuai dengan apa yang diharapkan.</p>	

e. Non-fungsional: *Operability*

Tujuan uji non-fungsional ini adalah untuk menguji kemudahan dalam mengoperasikan aplikasi yang telah dibuat. Uji akan yang dilakukan terhadap posisi atau letak tombol perintah, kolom-kolom dan teks yang terdapat pada *form*. Uji dapat dilihat pada Tabel 4.37.

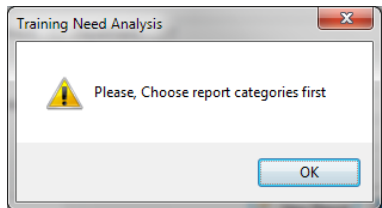
Tabel 4.37 Hasil Uji Non-Fungsional *Operability* Pada Fungsional Membuat Standar Kompetensi Jabatan

Keterangan	Hasil Sistem
Bahasa digunakan adalah bahasa yang disesuaikan dengan kebutuhan perusahaan.	
Tata letak keterangan maupun tombol-tombol perintah diposisikan secara rapi dan dikelompokkan berdasarkan fungsinya.	
Kesimpulan	
<p>Uji non-fungsional ini menyimpulkan bahwa posisi keterangan maupun perintah-perintah yang telah tersedia dikelompokkan sesuai fungsinya dan bahasa yang digunakan telah disesuaikan dengan kebutuhan perusahaan. Hal tersebut sudah dapat memudahkan pengguna dalam mengoperasikan aplikasi yang telah dibuat.</p>	

3. Melaksanakan Perencanaan Pelatihan

Mengenai uji fungsional melaksanakan perencanaan pelatihan lebih detil dapat dilihat pada Tabel 4.38.

Tabel 4.38 Hasil Uji Fungsional Melaksanakan Perencanaan Pelatihan

Fungsional	Melaksanakan Perencanaan Pelatihan	
Stakeholder	<i>Training officer</i>	
Alur Normal		
Aksi Pengguna	Respon Sistem	Hasil
Pengguna melihat laporan peserta pelatihan memilih sub menu “ <i>Training Participants</i> ” pada menu “ <i>Report</i> ”.	Sistem akan menampilkan <i>form</i> “ <i>Training Participants</i> ”.	Untuk hasilnya dapat dilihat pada Gambar 4.22.
Pengguna memilih kategori laporan yang akan ditampilkan.	Sistem akan menampilkan laporan peserta pelatihan.	Untuk hasilnya dapat dilihat pada Gambar 4.23.
Pengguna melihat laporan prioritas pelatihan dengan memilih sub menu “ <i>Priority Training Need Report</i> ” pada menu “ <i>Report</i> ”.	Sistem akan menampilkan <i>form</i> “ <i>Priority Training Need Report</i> ”.	Untuk hasilnya dapat dilihat pada Gambar 4.24.
Pengguna menentukan periode tahun kegiatan pelatihan yang akan dilaksanakan.	Sistem akan menampilkan laporan prioritas dalam melaksanakan kegiatan pelatihan.	Untuk hasilnya dapat dilihat pada Gambar 4.25.
Alur Alternatif		
Aksi Pengguna	Respon Sistem	Hasil
-	-	-
Alur Eksepsi		
Aksi Pengguna	Respon Sistem	Hasil
Pengguna tidak dapat melihat laporan peserta yang akan mengikuti pelatihan.	Sistem akan menampilkan pesan bahwa pengguna belum menentukan kategori laporan yang ditampilkan.	

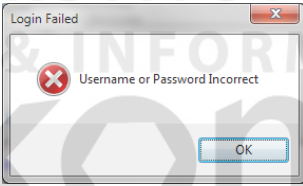
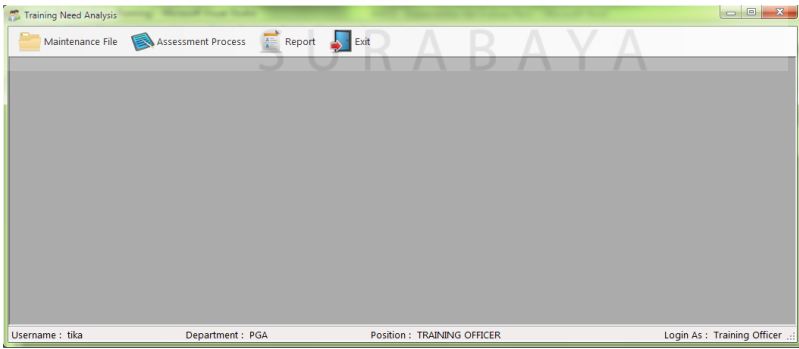
Kondisi Akhir	
Respon Sistem	Hasil
Sistem berhasil menampilkan laporan peserta yang akan mengikuti pelatihan.	Untuk hasilnya dapat dilihat pada Gambar 4.23.
Sistem berhasil menampilkan laporan prioritas kebutuhan pelatihan yang akan dilaksanakan.	Untuk hasilnya dapat dilihat pada Gambar 4.25.

Langkah selanjutnya adalah uji non-fungsional. Berikut adalah hasil uji non-fungsional pada fungsional melaksanakan perencanaan pelatihan:

a. Non-fungsional: *Security*

Tujuan dari uji non-fungsional ini adalah untuk menguji keamanan dari aplikasi yang telah dibuat. Hasil uji dapat dilihat pada Tabel 4.39.

Tabel 4.39 Hasil Uji Non-Fungsional *Security* Pada Fungsional Melaksanakan Perencanaan Pelatihan

Keterangan	Hasil Sistem
Pengguna sebagai <i>Training Officer</i> salah memasukkan <i>username</i> atau <i>password</i> .	
Otentikasi sistem terhadap <i>Training Officer</i> .	
Kesimpulan	
<p>Dari hasil uji non-fungsional <i>security</i>, dapat disimpulkan bahwa setelah pengguna melakukan <i>login</i>, sistem telah melakukan otentikasi yang sudah ditentukan. <i>Training Officer</i> dapat mengakses menu "<i>Maintenance File</i>", "<i>Assessment Process</i>" dan "<i>Report</i>".</p>	

b. Non-fungsional: *Correctness*

Tujuan dari non-fungsional ini adalah menampilkan laporan peserta yang akan mengikuti pelatihan sesuai dengan kebutuhan pelatihan yang telah ditentukan sebelumnya. Selain itu, sistem juga harus mampu menampilkan prioritas pelaksanaan pelatihan karyawan sesuai dengan aturan yang telah ditetapkan. Hasil uji dapat dilihat pada Tabel 4.40.

Tabel 4.40 Hasil Uji Non-Fungsional *Correctness* Pada Fungsional Melaksanakan Perencanaan Pelatihan

Keterangan	Hasil Sistem
Sistem berhasil menampilkan laporan peserta yang akan mengikuti pelatihan.	Untuk hasilnya dapat dilihat pada Gambar 4.23.
Sistem berhasil menampilkan laporan prioritas kebutuhan pelatihan pada tahun pelaksanaannya dan sesuai dengan aturan yang telah ditetapkan.	Untuk hasilnya dapat dilihat pada Gambar 4.25.
Kesimpulan	
Dari hasil uji coba coba ini dapat disimpulkan bahwa sistem berhasil menampilkan laporan peserta yang akan Laporan tersebut ditampilkan berdasarkan nama program pelatihan yang dipilih. Selain itu, laporan prioritas kebutuhan pelatihan juga berhasil ditampilkan untuk direncanakan pelaksanaannya pada tahun berikutnya.	

c. Non-fungsional: *Interface*

Tujuan uji non-fungsional ini adalah untuk melakukan pengujian pada desain antarmuka. Desain antarmuka mengacu pada antarmuka *windows* dan didukung dengan *icon-icon* pada semua jenis perintah. Adapun hasil ujinya dapat dilihat pada Tabel 4.41.

Tabel 4.41 Hasil Uji Non-Fungsional *Interface* Pada Fungsional Melaksanakan Perencanaan Pelatihan

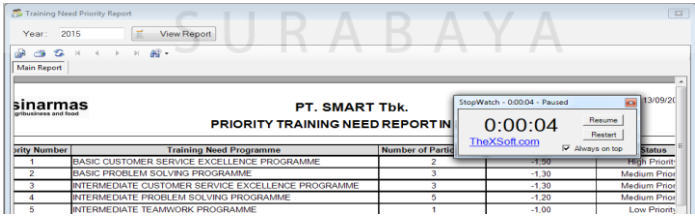
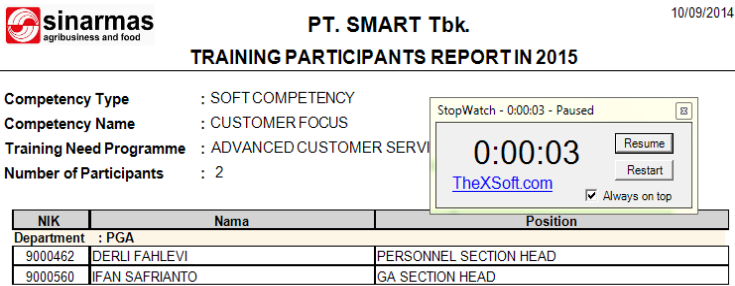
Keterangan	Hasil Sistem
Warna yang digunakan pada <i>form</i> tidak terlalu mencolok dan <i>icon-icon</i> yang digunakan pada jenis perintah menarik dan mudah dalam mendefinisikan perintah.	Untuk hasilnya dapat dilihat pada Gambar 4.22 dan Gambar 4.24.
Kesimpulan	
Sistem dirancang dengan desain yang disesuaikan dengan kebutuhan pengguna, sehingga dapat memberikan kemudahan dan kenyamanan dalam menjalankan aplikasi ini.	

d. Non-fungsional: *Performance*

Tujuan uji non-fungsional ini adalah untuk menguji kinerja dari aplikasi.

Untuk menampilkan laporan, aplikasi harus mampu melaksanakannya dalam waktu tidak lebih dari 30 detik. Adapun hasil ujinya dapat dilihat pada Tabel 4.42.

Tabel 4.42 Hasil Uji Non-Fungsional *Performance* Pada Fungsional Melaksanakan Perencanaan Pelatihan

Keterangan	Hasil Sistem																								
Waktu yang dibutuhkan dalam menampilkan laporan prioritas kebutuhan pelatihan yang dibaca dari <i>database</i> adalah 4 detik.	 <table border="1"> <thead> <tr> <th>Party Number</th> <th>Training Need Programme</th> <th>Number of Participants</th> <th>Priority</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>BASIC CUSTOMER SERVICE EXCELLENCE PROGRAMME</td> <td>2</td> <td>High Priority</td> </tr> <tr> <td>2</td> <td>BASIC PROBLEM SOLVING PROGRAMME</td> <td>3</td> <td>Medium Priority</td> </tr> <tr> <td>3</td> <td>INTERMEDIATE CUSTOMER SERVICE EXCELLENCE PROGRAMME</td> <td>3</td> <td>Medium Priority</td> </tr> <tr> <td>4</td> <td>INTERMEDIATE PROBLEM SOLVING PROGRAMME</td> <td>5</td> <td>Medium Priority</td> </tr> <tr> <td>5</td> <td>INTERMEDIATE TEAMWORK PROGRAMME</td> <td>1</td> <td>Low Priority</td> </tr> </tbody> </table>	Party Number	Training Need Programme	Number of Participants	Priority	1	BASIC CUSTOMER SERVICE EXCELLENCE PROGRAMME	2	High Priority	2	BASIC PROBLEM SOLVING PROGRAMME	3	Medium Priority	3	INTERMEDIATE CUSTOMER SERVICE EXCELLENCE PROGRAMME	3	Medium Priority	4	INTERMEDIATE PROBLEM SOLVING PROGRAMME	5	Medium Priority	5	INTERMEDIATE TEAMWORK PROGRAMME	1	Low Priority
Party Number	Training Need Programme	Number of Participants	Priority																						
1	BASIC CUSTOMER SERVICE EXCELLENCE PROGRAMME	2	High Priority																						
2	BASIC PROBLEM SOLVING PROGRAMME	3	Medium Priority																						
3	INTERMEDIATE CUSTOMER SERVICE EXCELLENCE PROGRAMME	3	Medium Priority																						
4	INTERMEDIATE PROBLEM SOLVING PROGRAMME	5	Medium Priority																						
5	INTERMEDIATE TEAMWORK PROGRAMME	1	Low Priority																						
Waktu yang dibutuhkan dalam menampilkan laporan peserta pelatihan yang dibaca dari <i>database</i> adalah 3 detik.	 <p>PT. SMART Tbk. 10/09/2014</p> <p>TRAINING PARTICIPANTS REPORT IN 2015</p> <p>Competency Type : SOFT COMPETENCY Competency Name : CUSTOMER FOCUS Training Need Programme : ADVANCED CUSTOMER SERVI Number of Participants : 2</p> <table border="1"> <thead> <tr> <th>NIK</th> <th>Nama</th> <th>Position</th> </tr> </thead> <tbody> <tr> <td>9000462</td> <td>DERLI FAHLEVI</td> <td>PERSONNEL SECTION HEAD</td> </tr> <tr> <td>9000560</td> <td>IFAN SAFRIANTO</td> <td>GA SECTION HEAD</td> </tr> </tbody> </table>	NIK	Nama	Position	9000462	DERLI FAHLEVI	PERSONNEL SECTION HEAD	9000560	IFAN SAFRIANTO	GA SECTION HEAD															
NIK	Nama	Position																							
9000462	DERLI FAHLEVI	PERSONNEL SECTION HEAD																							
9000560	IFAN SAFRIANTO	GA SECTION HEAD																							

Kesimpulan
Uji non-fungsional ini dilakukan dengan menggunakan komputer yang memiliki spesifikasi lebih rendah dari kebutuhan minimum perangkat lunak yang disarankan. Hasil pengujian ini menyimpulkan bahwa sistem mampu menampilkan laporan peserta yang akan mengikuti pelatihan dan laporan prioritas kebutuhan pelatihan yang dibaca dari <i>database</i> sesuai dengan apa yang diharapkan.

e. Non-fungsional: *Operability*

Tujuan uji non-fungsional ini adalah untuk menguji kemudahan dalam mengoperasikan aplikasi yang telah dibuat. Uji akan yang dilakukan terhadap posisi atau letak tombol perintah, kolom-kolom dan teks yang terdapat pada *form*. Uji dapat dilihat pada Tabel 4.43.

Tabel 4.43 Hasil Uji Non-Fungsional *Operability* Pada Fungsional Melaksanakan Perencanaan Pelatihan

Keterangan	Hasil Sistem
Bahasa digunakan adalah bahasa yang disesuaikan dengan kebutuhan perusahaan.	Untuk hasilnya dapat dilihat pada Gambar 4.22 dan Gambar 4.24.
Tata letak keterangan maupun tombol-tombol perintah diposisikan secara rapi dan dikelompokkan berdasarkan fungsinya.	Untuk hasilnya dapat dilihat pada Gambar 4.22 dan Gambar 4.24.
Kesimpulan	
Uji non-fungsional ini menyimpulkan bahwa posisi keterangan maupun perintah-perintah yang telah tersedia dikelompokkan sesuai fungsinya dan bahasa yang digunakan telah disesuaikan dengan kebutuhan perusahaan. Hal tersebut sudah dapat memudahkan pengguna dalam mengoperasikan aplikasi yang telah dibuat.	

4.5 Evaluasi

Setelah tahapan implementasi, uji coba fungsional dan non-fungsional dilakukan, selanjutnya adalah melakukan evaluasi terhadap sistem tersebut secara keseluruhan. Evaluasi dilakukan dengan membandingkan antara hasil *output*

program yaitu penentuan kebutuhan pelatihan karyawan dengan penentuan kebutuhan pelatihan tanpa menggunakan aplikasi.

Dalam proses penentuan kebutuhan pelatihan karyawan, dibutuhkan data penilaian kompetensi yang dimiliki oleh karyawan saat ini. Penilaian tersebut dilakukan pada karyawan yang terdapat di *Personnel and General Affair Department*. Sampel data penilaian tersebut diperoleh pada saat implementasi yang telah dilakukan di perusahaan. Sampel data penilaian dapat dilihat di Lampiran 11.

4.5.1 Penentuan Kebutuhan Pelatihan Dengan Aplikasi

Analisis kebutuhan pelatihan bagi karyawan menggunakan metode analisis kesenjangan antara tingkat kompetensi jabatan dengan tingkat kompetensi karyawan yang diimplementasikan ke dalam sebuah aplikasi ini dilakukan dengan cara menentukan kebutuhan pelatihan pada data penilaian kompetensi karyawan di *Personnel & General Affair Department*. Dari data penilaian tersebut akan dihasilkan tingkat kompetensi karyawan untuk saat ini. Kemudian dilakukan perhitungan kesenjangan antara tingkat kompetensi jabatan dengan tingkat kompetensi karyawan. Hasil dari perhitungan tersebut akan menentukan kebutuhan pelatihan yang harus diberikan pada karyawan. Kebutuhan pelatihan yang telah ditentukan dapat dibuat laporannya. Adapun laporan analisis kebutuhan pelatihan dapat dilihat pada Gambar 4.26.

NIK	Name	Department	Cost Center	Position
5010545	AKHMAD HARIYANTO	PGA	R124GNA	TIME ATTENDANT
SOFT COMPETENCY	CUSTOMER FOCUS	ADVANCED CUSTOMER SERVICE EXCELLENCE PROGRAMME		
SOFT COMPETENCY	PROBLEM SOLVING	INTERMEDIATE PROBLEM SOLVING PROGRAMME		
9000482	DERLI FAHLEVI	PGA	R124GNA	PERSONNEL SECTION HEAD
SOFT COMPETENCY	CUSTOMER FOCUS	INTERMEDIATE CUSTOMER SERVICE EXCELLENCE PROGRAMME		
11001242	FITRI AMELIA	PGA	R124GNA	PERSONNEL ADMINISTRATION
SOFT COMPETENCY	PROBLEM SOLVING	BASIC PROBLEM SOLVING PROGRAMME		
SOFT COMPETENCY	TEAM LEADERSHIP	BASIC LEADERSHIP PROGRAMME		
9000580	IFAN SAFRIANTO	PGA	R124GNA	GA SECTION HEAD
SOFT COMPETENCY	CUSTOMER FOCUS	INTERMEDIATE CUSTOMER SERVICE EXCELLENCE PROGRAMME		
5010728	IRJANI	PGA	R124GNA	TIME ATTENDANT
SOFT COMPETENCY	PROBLEM SOLVING	INTERMEDIATE PROBLEM SOLVING PROGRAMME		
SOFT COMPETENCY	TEAM LEADERSHIP	INTERMEDIATE LEADERSHIP PROGRAMME		
SOFT COMPETENCY	TEAMWORK	INTERMEDIATE TEAMWORK PROGRAMME		
5010759	JUN ROBUSTANA HADI	PGA	R124GNA	PERSONNEL ADMINISTRATION
SOFT COMPETENCY	CUSTOMER FOCUS	BASIC CUSTOMER SERVICE EXCELLENCE PROGRAMME		
SOFT COMPETENCY	PROBLEM SOLVING	BASIC PROBLEM SOLVING PROGRAMME		
SOFT COMPETENCY	TEAM LEADERSHIP	BASIC LEADERSHIP PROGRAMME		
5010788	KARMELIA JULIANI	PGA	R124GNA	RECRUITMENT OFFICER
SOFT COMPETENCY	PROBLEM SOLVING	INTERMEDIATE PROBLEM SOLVING PROGRAMME		
SOFT COMPETENCY	TEAM LEADERSHIP	INTERMEDIATE LEADERSHIP PROGRAMME		
5010830	MEME MEGAWATI KURNIADI	PGA	R124GNA	PAYROLL OFFICER
SOFT COMPETENCY	CUSTOMER FOCUS	ADVANCED CUSTOMER SERVICE EXCELLENCE PROGRAMME		
SOFT COMPETENCY	PROBLEM SOLVING	INTERMEDIATE PROBLEM SOLVING PROGRAMME		
5010801	MOCH. ARIFIN	PGA	R124GNA	EHFS SECTION HEAD
SOFT COMPETENCY	CUSTOMER FOCUS	INTERMEDIATE CUSTOMER SERVICE EXCELLENCE PROGRAMME		
12000752	MUSTIKA INDAH WARDANI	PGA	R124GNA	TRAINING OFFICER

Gambar 4.26 Laporan Analisis Kebutuhan Pelatihan Tahun 2014

Dari Gambar 4.26 dapat disimpulkan bahwa analisis kebutuhan pelatihan menghasilkan jenis-jenis program pelatihan yang harus diikuti oleh masing-masing karyawan untuk periode tahun ke depan dari perencanaan pelatihan tahun 2014. Untuk mengetahui program pelatihan yang akan dilaksanakan sesuai dengan kebutuhan beserta prioritas pelaksanaannya dapat dilihat di laporan prioritas kebutuhan pelatihan pada Gambar 4.27. Laporan tersebut menghasilkan prioritas pelaksanaan pelatihan untuk periode ke depan, yaitu tahun 2015 yang berasal dari laporan analisis kebutuhan pelatihan tahun 2014.

Priority Number	Training Need Programme	Number of Participants	Average Gap	Status
1	BASIC CUSTOMER SERVICE EXCELLENCE PROGRAMME	2	-1,50	High Priority
2	BASIC PROBLEM SOLVING PROGRAMME	3	-1,30	Medium Priority
3	INTERMEDIATE CUSTOMER SERVICE EXCELLENCE PROGRAMME	3	-1,30	Medium Priority
4	INTERMEDIATE PROBLEM SOLVING PROGRAMME	5	-1,20	Medium Priority
5	INTERMEDIATE TEAMWORK PROGRAMME	1	-1,00	Low Priority
6	ADVANCED CUSTOMER SERVICE EXCELLENCE PROGRAMME	2	-1,00	Low Priority
7	BASIC LEADERSHIP PROGRAMME	2	-1,00	Low Priority
8	INTERMEDIATE LEADERSHIP PROGRAMME	3	-1,00	Low Priority

Gambar 4.27 Laporan Prioritas Kebutuhan Pelatihan Tahun 2015

4.5.2 Penentuan Kebutuhan Pelatihan Tanpa Aplikasi

Sebelum terdapat aplikasi analisis kebutuhan pelatihan, perusahaan melakukan penentuan kebutuhan pelatihan pada masing-masing karyawan dengan cara mengajukan jenis pelatihan kepada pihak *training officer*. Pengajuan dilakukan oleh pihak *department head* pada masing-masing departemen. *Department Head* mengajukan kebutuhan pelatihan tanpa identifikasi antara kebutuhan kompetensi yang seharusnya dimiliki oleh karyawan dengan kompetensi yang dimiliki karyawan saat ini. Sehingga, *department head* dapat bebas melakukan pengajuan jenis pelatihan dengan tidak mepedulikan apakah pelatihan sesuai dengan kebutuhan karyawan atau tidak. Berikut Tabel 4.44 merupakan contoh pengajuan kebutuhan pelatihan yang dilakukan oleh *Production Department Head* pada tahun 2012.

Tabel 4.44 Pengajuan Kebutuhan Pelatihan *Production Department* Tahun 2012

Nama	Jabatan	Cost Center	Jenis Training
Dimas Vanguardy	Fractionation Sec. Head	R124FRA	Manufacturing Process & Equipment
Dimas Vanguardy	Fractionation Sec. Head	R124FRA	SAP R/3 V4.6C
Moh. Hadi	Fractionation Officer	R124FRA	Stress Management
Moh. Hadi	Fractionation Officer	R124FRA	Supervisory Skill
Warsito	Fractionation Officer	R124FRA	SAP R/3 V4.6C
Tri Sumarno	Fractionation Officer	R124FRA	Stress Management
Tri Sumarno	Fractionation Officer	R124FRA	Supervisory Skill
Dewi Setyowati	Filling Officer	R124FIL	Industrial Hazard Control
Bambang SIP	Filling Shift Leader	R124FIL	Effective Supervisory Skill

Nama	Jabatan	Cost Center	Jenis Training
Gunawan	Filling Shift Leader	R124FIL	Effective Supervisory Skill
Suharto	Filling Shift Leader	R124FIL	Effective Supervisory Skill

4.5.3 Perbandingan Hasil Evaluasi

Dari hasil uji coba melalui aplikasi analisis kebutuhan pelatihan dan melalui contoh penentuan kebutuhan pelatihan secara manual, proses penentuan kebutuhan pelatihan yang lebih efektif adalah dengan menggunakan aplikasi. Proses penentuan kebutuhan pelatihan dengan menggunakan aplikasi dapat melakukan analisis kebutuhan pelatihan berdasarkan kesenjangan antara kompetensi yang seharusnya dimiliki karyawan dengan kompetensi yang dimiliki karyawan saat ini. Sehingga jenis pelatihan yang akan diberikan sesuai dengan kebutuhan karyawan yang sebenarnya.

Dengan adanya keefektifan pada proses penentuan kebutuhan pelatihan, maka dipastikan bahwa biaya kebutuhan pelatihan yang dikeluarkan tidak akan sia-sia karena kebutuhan pelatihan diberikan kepada karyawan secara tepat serta sesuai dengan kebutuhannya untuk meningkatkan kompetensi. Selain itu, penggunaan aplikasi juga berpengaruh pada penghematan biaya kebutuhan pelatihan. Penghematan biaya akan berdampak jika terdapat perbandingan penggunaan aplikasi dengan penentuan kebutuhan pelatihan yang dilakukan oleh *department head* saat mengikutsertakan banyak karyawan pada suatu kegiatan pelatihan tanpa identifikasi kebutuhan sebenarnya.